

DUAL-PIVOT QUICKSORT: OPTIMALITY, ANALYSIS AND ZEROS OF ASSOCIATED LATTICE PATHS

MARTIN AUMÜLLER, MARTIN DIETZFELBINGER, CLEMENS HEUBERGER,
DANIEL KRENN, AND HELMUT PRODINGER

ABSTRACT. We present an average case analysis of a variant of dual-pivot quicksort. We show that the used algorithmic partitioning strategy is optimal, i.e., it minimizes the expected number of key comparisons. For the analysis, we calculate the expected number of comparisons exactly as well as asymptotically, in particular, we provide exact expressions for the linear, logarithmic, and constant terms.

An essential step is the analysis of zeros of lattice paths in a certain probability model. Along the way a combinatorial identity is proven.

1. INTRODUCTION

Dual-pivot quicksort [15, 18, 2] is a family of sorting algorithms related to the well-known quicksort algorithm. In order to sort an input sequence (a_1, \dots, a_n) of distinct elements, dual-pivot quicksort algorithms work as follows. (For simplicity we forbid repeated elements in the input.) If $n \leq 1$, there is nothing to do. If $n \geq 2$, two of the input elements are selected as pivots. Let p be the smaller and q be the larger pivot. The next step is to partition the remaining elements into

- the elements smaller than p (“small elements”),
- the elements between p and q (“medium elements”), and
- the elements larger than q (“large elements”).

Then the procedure is applied recursively to these three groups to complete the sorting.

The cost measure used in this work is the number of comparisons between elements. As is common, we will assume the input sequence is in random order, which means that each permutation of the n elements occurs with probability $1/n!$. With this assumption we may, without loss of generality, choose a_1 and a_n as the pivots. Even in this setting there are different dual-pivot quicksort algorithms; their difference lies in the way the partitioning is organized, which influences the partitioning cost. This is in contrast to standard quicksort with one pivot, where the partitioning strategy does not influence the cost—in partitioning always one comparison is needed per non-pivot element. In dual-pivot quicksort, the average

2010 *Mathematics Subject Classification.* 05A16, 68R05, 68P10, 68Q25, 68W40.

Key words and phrases. Dual-pivot quicksort, lattice paths, asymptotic enumeration, combinatorial identity.

C. Heuberger and D. Krenn are supported by the Austrian Science Fund (FWF): P 24644-N26 and by the Karl Popper Kolleg “Modeling–Simulation–Optimization” funded by the Alpen-Adria-Universität Klagenfurt and by the Carinthian Economic Promotion Fund (KWF).

H. Prodinger is supported by an incentive grant of the National Research Foundation of South Africa.

cost (over all permutations) of partitioning and of sorting can be analyzed only when the partitioning strategy is fixed.

Only in 2009, Yaroslavskiy, Bentley, and Bloch [19] described a dual-pivot quicksort algorithm that makes $1.9n \log n + O(n)$ key comparisons [18].¹ This beats the classical quicksort algorithm [7], which needs $2(n+1)H_n - 4n = 2n \log n + O(n)$ comparisons on average. (Here H_n denote the harmonic numbers. We refer to the book [8, 5.2.2 (24)] for this asymptotic as well as an exact result.) Wild’s Ph.D. thesis [17] discusses implications of the Yaroslavskiy–Bentley–Bloch-algorithm and many variants in detail.

In [2], the first two authors of this article described the full design space for dual-pivot quicksort algorithms with respect to counting element comparisons. Among others, they studied a special partitioning strategy called “Count”; it makes its decision on the number of small and large elements seen at a certain point. It was shown that the dual-pivot quicksort carries out $1.8n \log n + O(n)$ key comparisons on average when this partitioning strategy is used and that the main term in this formula is optimal/minimal. (They showed that no other partitioning strategy can have a lower main term, even if this strategy has access to a certain oracle, cf. their strategy “Clairvoyant”.)

One purpose of this paper is to make the expected number of comparisons in the algorithm “Count” precise, both for partitioning and for the resulting dual-pivot quicksort variants. Moreover, we will show that “Count” is optimal among all algorithmic strategies, see Part III for details.

Already in [2] it was noted that the exact value of the expected partitioning cost (i.e., the number of comparisons) of the mentioned strategy depends on the expected number of the zeros of certain lattice paths (Part I). A complete understanding of this situation is the basis for our analysis of dual-pivot quicksort, which appears in Part II.

Lattice path enumeration has a long tradition. An early reference is [10]; a recent survey paper on the subject is [9].

2. OVERVIEW AND RESULTS

This work is split into three parts. We give a brief overview on the main results of each of these parts here. We use the Iversonian expression

$$[expr] = \begin{cases} 1 & \text{if } expr \text{ is true,} \\ 0 & \text{if } expr \text{ is false,} \end{cases}$$

popularized by Graham, Knuth, and Patashnik [4].

The harmonic numbers and variants will be denoted by

$$H_n = \sum_{m=1}^n \frac{1}{m}, \quad H_n^{\text{odd}} = \sum_{m=1}^n \frac{[m \text{ odd}]}{m} \quad \text{and} \quad H_n^{\text{alt}} = \sum_{m=1}^n \frac{(-1)^m}{m}.$$

Of course, there are relations between these three definitions such as $H_n^{\text{alt}} = H_n - 2H_n^{\text{odd}}$ and $H_n^{\text{odd}} + H_{\lfloor n/2 \rfloor} = H_n$, but it will turn out to be much more convenient to use all three notations.

¹In this paper “log” denotes the natural logarithm to base e .

We denote multinomial coefficients by

$$\binom{n}{c_1, c_2, \dots, c_\delta} = \frac{n!}{c_1! c_2! \dots c_\delta!}.$$

Part I: Lattice Paths. In the first part we analyze certain lattice paths of a fixed length n . We start at the origin $(0, 0)$, allow steps $(1, +1)$ and $(1, -1)$ and stop after n steps. To be precise, the ending point is chosen from the set $\{(n, -n), (n, -n+2), \dots, (n, n-2), (n, n)\}$ of feasible points uniformly at random. Once this ending point is fixed, all paths from the origin to this particular ending point are equally likely. We are interested in the number of zeros, denoted by the random variable Z_n , of such paths.

An exact formula for the expected number $\mathbb{E}(Z_n)$ of zeros is derived in two different ways (see identity (2.1) for these formulæ): On the one hand, we use the symbolic method and generating functions (see Section 5), which gives the result in form of a double sum (Theorem 5.1). This machinery extends well to higher moments and also allows us to obtain the distribution. The exact distribution is given in Theorem 9.1; its limiting behavior as $n \rightarrow \infty$ is the discrete distribution

$$\mathbb{P}(Z_n = r) \sim \frac{1}{r(r+1)}.$$

On the other hand, a more probabilistic approach gives the expectation $\mathbb{E}(Z_n)$ as the simple single sum

$$\mathbb{E}(Z_n) = \sum_{m=1}^{n+1} \frac{[m \text{ odd}]}{m} = H_{n+1}^{\text{odd}},$$

see Section 6 for more details. The asymptotic behavior $\mathbb{E}(Z_n) \sim \frac{1}{2} \log n$ can be extracted (Section 8).

The two approaches above give rise to the identity

$$\sum_{m=1}^{n+1} \frac{[m \text{ odd}]}{m} = \frac{4}{n+1} \sum_{0 \leq k < \ell < \lceil n/2 \rceil} \frac{\binom{n}{k}}{\binom{n}{\ell}} + [n \text{ even}] \frac{1}{n+1} \left(\frac{2^n}{\binom{n}{n/2}} - 1 \right) + 1; \quad (2.1)$$

the double sum above equals the single sum of Theorem 6.1 by combinatorial considerations. One might ask about a direct proof of this interesting combinatorial identity. This can be achieved by methods related to hypergeometric sums and the computational proof is presented in Section 7. We also provide a completely elementary proof which is “purely human”.

Part II: Dual-Pivot Quicksort. One main result of this work analyzes key comparisons in the dual-pivot quicksort algorithm that uses the optimal (see Part III) partitioning strategy “Count”. Aumüller and Dietzfelbinger showed in [2] that this algorithm (as well as the optimal, but unrealistic as oracle-based strategy “Clairvoyant”) requires $1.8n \log n + O(n)$ comparisons on average, which improves on the average number of comparisons in quicksort ($2n \log n + O(n)$) and the recent dual-pivot algorithm of Yaroslavskiy et al. ($1.9n \log n + O(n)$, see [18]). However, for real-world input sizes n the (usually negative) factor in the linear term has a great influence on the comparison count. Our asymptotic result is stated as the following theorem.

Theorem. *The average number of comparisons in the dual-pivot quicksort algorithm with a comparison-optimal partitioning strategy is*

$$\frac{9}{5}n \log n + An + B \log n + C + O(1/n)$$

as $n \rightarrow \infty$, with

$$A = \frac{9}{5}\gamma + \frac{1}{5}\log 2 - \frac{89}{25} = -2.382\dots$$

The constants B and C are explicitly given, too, and more terms of the asymptotics are presented. The precise result is formulated as Corollary 12.2.

In fact, we even get an exact expression for the average comparison cost. The precise result is formulated as Theorem 12.1. The same analysis can be carried out for the non-algorithmic—it has access to an oracle—partitioning strategy “Clairvoyant” [2], see Appendix B for the result.

Part III: Optimality of the Strategy “Count”. In Aumüller and Dietzfelbinger [2] it was shown that the strategy “Clairvoyant”, which has access to an oracle to predict the number of small and large elements in the remaining list, minimizes the number of key comparisons among all such strategies with oracle; thus it is called optimal. The main result of Part III is that the algorithmic partitioning strategy “Count” is an optimal strategy among all algorithmic dual-pivot partitioning strategies. Therefore we can show a sharp lower bound for the average number of key comparisons of arbitrary dual-pivot quicksort algorithms (Theorem 15.2).

3. BACKGROUND: LATTICE PATHS IN \mathbb{N}_0^h

In this paper, we will twice need lattice paths with very similar properties. We collect the relevant definitions and results in this section.

Let $h \geq 1$ be an integer and $n \geq 0$. For $0 \leq t \leq n$, we set $\Omega_t := \{c \in \mathbb{N}_0^h \mid c_1 + \dots + c_h = t\}$. In this section, we consider lattice paths in \mathbb{N}_0^h . All lattice paths of this section start in the origin and the admissible steps are $e_1 = (1, 0, \dots, 0)^\top, \dots, e_h = (0, \dots, 0, 1)^\top$. For any lattice path v with n steps and $0 \leq t \leq n$, we write $v_{\leq t}$ for the lattice path consisting of the first t steps of v . By construction $v_{\leq t}$ ends in Ω_t .

We consider a random lattice path V with n steps under the following probability model:

- All endpoints in Ω_n are equally likely.
- For $c \in \Omega_n$, all lattice paths ending in c are equally likely.

Lemma 3.1. *For $0 \leq t \leq n$ and $c \in \Omega_t$, we have*

$$\mathbb{P}(V_{\leq t} \text{ ends in } c) = \frac{1}{|\Omega_t|} = \frac{1}{\binom{t+h-1}{h-1}},$$

$$\mathbb{P}(V_{\leq t+1} \text{ ends in } c + e_j \mid V_{\leq t} \text{ ends in } c) = \frac{c_j + 1}{t + h} \quad \text{for } t < n,$$

where as above, $V_{\leq t}$ denotes the initial segment of length t of V .

Proof. We use another random walk V' such that for $0 \leq t < n$,

$$\mathbb{P}(V'_{\leq t+1} \text{ ends in } c + e_j \mid V'_{\leq t} \text{ ends in } c) = \frac{c_j + 1}{t + h}. \quad (3.1)$$

We investigate V' with the aim of proving that V and V' are identically distributed.

Let $0 \leq t \leq n$ and v' be some lattice path with n steps. Assume that $v'_{\leq t}$ ends in $c \in \Omega_t$. We claim that

$$\mathbb{P}(V'_{\leq t} = v'_{\leq t}) = \frac{1}{\binom{t+h-1}{h-1}} \frac{1}{\binom{t}{c_1, c_2, \dots, c_h}}. \quad (3.2)$$

We prove this by induction on t . This is certainly true for $t = 0$ because $\Omega_0 = \{0\}$. Let now e_k be the last step of v'_{t+1} . Then we have

$$\begin{aligned} \mathbb{P}(V'_{\leq t+1} = v'_{\leq t+1}) &= \mathbb{P}(V'_{\leq t+1} \text{ ends in } c + e_k \mid V'_{\leq t} \text{ ends in } c) \mathbb{P}(V'_{\leq t} = v'_{\leq t}) \\ &= \frac{c_k + 1}{t + h} \frac{1}{\binom{t+h-1}{h-1}} \frac{1}{\binom{t}{c_1, c_2, \dots, c_h}} \\ &= \frac{1}{\frac{t+h}{t+1} \binom{t+h-1}{t} \frac{t+1}{c_k+1} \binom{t}{c_1, c_2, \dots, c_h}} \\ &= \frac{1}{\binom{t+h}{t+1}} \frac{1}{\binom{t+1}{c_1, \dots, c_k+1, \dots, c_h}}, \end{aligned}$$

which is (3.2) for $t + 1$.

Note that (3.2) implies that $\mathbb{P}(V'_{\leq t} = v'_{\leq t})$ only depends on the endpoint c of $v'_{\leq t}$ and not on the initial segment $v'_{\leq t}$ itself. Thus all $v'_{\leq t}$ ending in c are equally likely. There are $\binom{t}{c_1, c_2, \dots, c_h}$ lattice paths of length t ending in c , thus (3.2) implies that

$$\mathbb{P}(V'_t \text{ ends in } c) = \frac{1}{\binom{t+h-1}{h-1}} = \frac{1}{|\Omega_t|}. \quad (3.3)$$

This probability does not depend on $c \in \Omega_t$, thus all endpoints c after t steps are equally likely.

For $t = n$, the last two observations imply that V' and V are identically distributed. Thus the assertions of the lemma follow from (3.3) and (3.1). \square

Note that the random walk V' considered in this proof is another formulation of a contagion Pólya urn model (also called Pólya–Eggenberger urn model) with balls of h colors. Initially, there is one ball of each color in the urn. In each step, a ball is drawn at random and is replaced by two balls of the same color. The color of the ball determines the next step of V' .

Part I. Lattice Paths

In this first part we analyze certain lattice paths with positive and negative ordinate values of a fixed length n . These are introduced in Section 4 by a precise description of the probabilistic model. We will work with this model throughout Part I, and we analyze the number of zeros Z_n .

The outline is as follows: We derive an exact expression for the expected number $\mathbb{E}(Z_n)$ of zeros by the generating functions machinery in Section 5 and then continue with a more probabilistic approach in Section 6. Asymptotic considerations follow (Section 8). Direct proofs of the obtained identity are given in Section 7 and the distribution of Z_n is tackled in Section 9.

4. PROBABILISTIC MODEL

We consider paths of a given length n on the lattice \mathbb{Z}^2 , where only *up-steps* $(1, +1)$ and *down-steps* $(1, -1)$ are allowed. These paths are chosen at random according to the rules below.

Let us fix a length $n \in \mathbb{N}_0$. A path² W_n starting in the origin $(0, 0)$ (no choice for this starting point) is chosen according to the following rules.

- First, we choose an ending point (n, D) where D is a random integer uniformly distributed in $\{-n, -n+2, \dots, n-2, n\}$, i. e., $D = d$ occurs only for integers d with $|d| \leq n$ and $d \equiv n \pmod{2}$.
- Second, a path is chosen uniformly at random among all paths from $(0, 0)$ to (n, D) .

The conditions on D characterize those ending points which are reachable from $(0, 0)$.

We are interested in the number of intersections with the horizontal axis of a path. To make this precise, we define a *zero* of a path W_n as a point $(x, 0) \in W_n$.

Thus, let W_n be a path of length n which is chosen according to the probabilistic model above and define the random variable

$$Z_n = \text{number of zeros of } W_n.$$

In the following sections, we determine the value of $\mathbb{E}(Z_n)$ exactly (Sections 5 and 6), as well as asymptotically (Section 8). In Section 5, we use the machinery of generating functions. This machinery turns out to be overkill if we are just interested in the expectation $\mathbb{E}(Z_n)$. However, it easily allows extension to higher moments and the limiting distribution.

In Section 6, we follow a probabilistic approach, which first gives a result on the probability model: the equidistribution at the final values turns out to carry over to every fixed length initial segment of the path. This result yields a very simple expression for the expectation $\mathbb{E}(Z_n)$ in terms of harmonic numbers, and thus immediately yields a precise asymptotic expansion for $\mathbb{E}(Z_n)$. This distributional result is a consequence of the results on more general lattice paths considered in Section 3. The generating function approach, however, gives the expectation in terms of a double sum of quotients of binomial coefficients (the right-hand side of (2.1)).

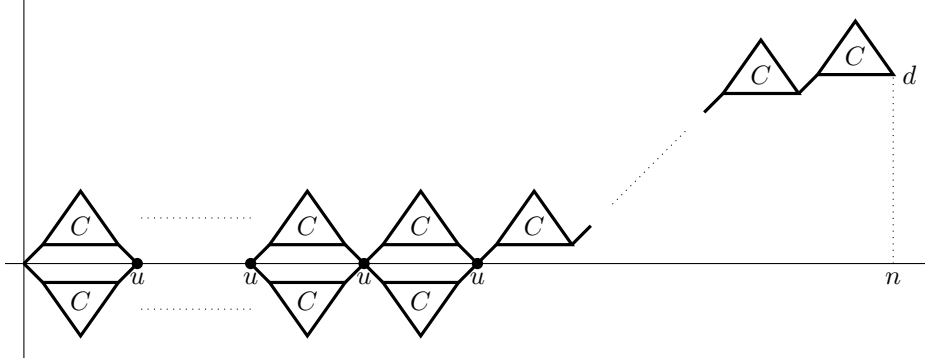
Section 7 gives a direct computational proof that these two results coincide. The original expression in [2] (a double sum over a quotient of a product of binomial coefficients and a binomial coefficient) is also shown to be equal to our identity, see Section 7. Explicit as well as asymptotic expressions for the distribution $\mathbb{P}(Z_n = r)$ can be found in Section 9.

5. USING THE GENERATING FUNCTION MACHINERY

Theorem 5.1. *For a randomly (as described in Section 4) chosen path of length n , the expected number of zeros is*

$$\mathbb{E}(Z_n) = \frac{4}{n+1} \sum_{0 \leq k < \ell < \lceil n/2 \rceil} \frac{\binom{n}{k}}{\binom{n}{\ell}} + [n \text{ even}] \frac{1}{n+1} \left(\frac{2^n}{\binom{n}{n/2}} - 1 \right) + 1.$$

²We denote the lattice paths by the symbol W_n (walks); the more natural symbol P_n is used later for the partitioning cost.

FIGURE 5.1. Decomposition of a lattice path for $s \geq 0$ marking zeros.

The remaining part of this section is devoted to the proof of this theorem. The main technique is to model the lattice paths by means of combinatorial classes and generating functions. For simplicity of notation, we denote a combinatorial class by its corresponding generating function.

Concerning the generating functions, we mark a step to the right (i.e., up- and down-steps) by the variable z and a zero (except the first) by u . Note that we do not mark the zero at $(0, 0)$ for technical reasons; we'll take this into account at the end by adding a 1 to the final result. Thus, the coefficient of $z^n u^{r-1}$ of the function $Q_d(z, u)$ (the generating function of all paths starting in $(0, 0)$ and ending in some (n, d)) equals the number of paths of length n that have exactly r zeros.

We also need the following auxiliary function. The generating function $C(z)$ counts all Catalan paths, i.e., paths starting and ending at the same height, but not going below it. This equals

$$C(z) = \frac{1 - \sqrt{1 - 4z^2}}{2z^2}.$$

In Figure 5.1, we give a schematic decomposition of a path from $(0, 0)$ to (n, d) for non-negative d . This decomposition translates to the following parts of the generating function (the path is read from the left to the right).

- We start at $(0, 0)$ by either doing a single ascent or a single decent (marked by a z), then continue with a $C(z)$ -block and do a single decent or ascent respectively (marked by a z as well) again. Thus, we are back at a zero.
- We mark this zero by the symbol u .
- We repeat such up/down blocks $zC(z)z$, each one preceded by a zero u , a finite number of times.
- We end by d consecutive blocks of $C(z)$, each preceded by a single ascent encoded as z . This gives the paths to their end point at height d .

If $d < 0$, then the construction is the same, but everything is reflected at the horizontal axis.

Continuing using the symbolic method—the description above is already part of it, see, for example, Flajolet and Sedgewick [3]—the decomposition above translates

to the generating function

$$Q_d(z, u) = \frac{C(z)^{|d|} z^{|d|}}{1 - 2uz^2 C(z)}, \quad (5.1)$$

which we will use from now on. Note that the coefficient 2 reflects the fact that there are two choices (up and down) for the blocks between zeros.

To obtain a nice explicit form, we perform a change of variables. The result is stated in the following lemma.

Lemma 5.2. *With the transformation $z = v/(1 + v^2)$ we have*

$$Q_d(z, u) = \frac{v^{|d|}(1 + v^2)}{1 - v^2(2u - 1)}.$$

Proof. Transforming the counting generating function of Catalan paths yields

$$C(z) = 1 + v^2.$$

Thus (5.1) becomes

$$Q_d(z, u) = (1 + v^2)^{|d|} \left(\frac{v}{1 + v^2} \right)^{|d|} \frac{1}{1 - 2u \left(\frac{v}{1 + v^2} \right)^2 (1 + v^2)},$$

which can be simplified to the expression stated in the lemma. \square

The next step is to extract the coefficients out of the expressions obtained in the previous lemma. First we rewrite the extraction of the coefficients from the “ z -world” to the “ v -world”, see Lemma 5.3. Afterwards, in Lemma 5.4, the coefficients can be determined quite easily.

Lemma 5.3. *Let $F(z)$ be an analytic function in a neighborhood of the origin. Then we have*

$$[z^n]F(z) = [v^n](1 - v^2)(1 + v^2)^{n-1} F\left(\frac{v}{1 + v^2}\right).$$

Proof. We use Cauchy’s formula to extract the coefficients of $F(z)$ as

$$[z^n]F(z) = \frac{1}{2\pi i} \oint_{\mathcal{C}} F(z) \frac{dz}{z^{n+1}}$$

where \mathcal{C} is a positively oriented small circle around the origin. Under the transformation $z = v/(1 + v^2)$, the circle \mathcal{C} is transformed to a contour \mathcal{C}' which still winds exactly once around the origin. Using Cauchy’s formula again, we obtain

$$\begin{aligned} [z^n]F(z) &= \frac{1}{2\pi i} \oint_{\mathcal{C}'} F\left(\frac{v}{1 + v^2}\right) \frac{(1 + v^2)^{n+1}}{v^{n+1}} \frac{1 - v^2}{(1 + v^2)^2} dv \\ &= [v^n](1 - v^2)(1 + v^2)^{n-1} F\left(\frac{v}{1 + v^2}\right). \end{aligned}$$

\square

Now we are ready to calculate the desired coefficients.

Lemma 5.4. *Suppose $n \equiv d \pmod{2}$. Then we have*

$$[z^n]Q_d(z, 1) = \binom{n}{(n-d)/2}$$

and, moreover,

$$[z^n] \frac{\partial}{\partial u} Q_d(z, u) \Big|_{u=1} = 2 \sum_{k=0}^{(n-|d|)/2-1} \binom{n}{k}.$$

Proof. As $n \equiv d \pmod{2}$, the number $n - d$ is even, and so we can set $\ell = \frac{1}{2}(n - d)$. Then $[z^n] Q_d(z, 1)$ is the number of paths from $(0, 0)$ to (n, d) . These paths have $n - \ell$ up-steps and ℓ down-steps; thus there are $\binom{n}{\ell}$ many such paths.

For the second part of the lemma, we restrict ourselves to $d \geq 0$ (otherwise use $-d$ and the symmetry in d of the generating function (5.1) instead). We start with the result of Lemma 5.2. Taking the first derivative and setting $u = 1$ yields

$$\frac{\partial}{\partial u} Q_d(z, u) \Big|_{u=1} = \frac{2v^{d+2}(1+v^2)}{(1-v^2)^2}.$$

Thus, by using Lemma 5.3, we get

$$[z^n] \frac{2v^{d+2}(1+v^2)}{(1-v^2)^2} = 2[v^{n-d-2}] \frac{(1+v^2)^n}{1-v^2}.$$

We use ℓ as above and get

$$[v^{n-d-2}] \frac{(1+v^2)^n}{1-v^2} = [v^{2\ell-2}] \frac{(1+v^2)^n}{1-v^2} = [v^{\ell-1}] \frac{(1+v)^n}{1-v} = \sum_{k=0}^{\ell-1} \binom{n}{k}$$

as desired. \square

We are now ready to prove the main theorem (Theorem 5.1) of this section, which provides an expression for the expected number of zeros. This exact expression is written as a double sum.

Proof of Theorem 5.1. By Lemma 5.4, the average number of zeros (except the zero at the origin) of a path of length n which ends in (n, d) is

$$\mu_{n,d} = \frac{[z^n] \frac{\partial}{\partial u} Q_d(z, u) \Big|_{u=1}}{[z^n] Q_d(z, 1)} = \frac{2}{\binom{n}{\ell}} \sum_{k=0}^{\ell-1} \binom{n}{k},$$

where we have set $\ell = \frac{1}{2}(n - |d|)$ as in the proof of Lemma 5.4. If $d = 0$, this simplifies to

$$\mu_{n,0} = \frac{2}{\binom{n}{n/2}} \sum_{k=0}^{n/2-1} \binom{n}{k} = \frac{2^n}{\binom{n}{n/2}} - 1. \quad (5.2)$$

If $n \not\equiv d \pmod{2}$, then we set $\mu_{n,d} = 0$.

Summing up yields

$$\begin{aligned} \sum_{d=-n}^n \mu_{n,d} &= 2 \sum_{d=1}^n \mu_{n,d} + \mu_{n,0} = 4 \sum_{\ell=0}^{\lceil n/2 \rceil - 1} \frac{1}{\binom{n}{\ell}} \sum_{k=0}^{\ell-1} \binom{n}{k} + \mu_{n,0} \\ &= 4 \sum_{0 \leq k < \ell < \lceil n/2 \rceil} \frac{\binom{n}{k}}{\binom{n}{\ell}} + [n \text{ even}] \left(\frac{2^n}{\binom{n}{n/2}} - 1 \right). \end{aligned}$$

Dividing by the number $n + 1$ of possible end points and adding 1 for the zero at the origin completes the proof of Theorem 5.1. \square

6. A PROBABILISTIC APPROACH

Theorem 6.1. *For a randomly (as described in Section 4) chosen path of length n , the expected number of zeros is*

$$\mathbb{E}(Z_n) = H_{n+1}^{\text{odd}}.$$

In the analysis of the quicksort algorithm in Part II, we need an *up-from-zero situation*, which is a point $(t, 0) \in W_n$ such that $(t+1, 1) \in W_n$. Define the random variable

$$Z_n^{\nearrow} = \text{number of up-from-zero situations of } W_n.$$

The following corollary provides the expected value of this random variable.

Corollary 6.2. *For a randomly (as described in Section 4) chosen path of length n , the expected number of up-from-zero situations is*

$$\mathbb{E}(Z_n^{\nearrow}) = \frac{1}{2} H_n^{\text{odd}}.$$

In order to prove the theorem and the corollary, we need the following property of our paths.

Lemma 6.3. *Let $t \in \mathbb{N}_0$ with $t \leq n$. The probability that a random path W_n (as defined in Section 4) runs through (t, k) is*

$$\mathbb{P}((t, k) \in W_n) = \frac{1}{t+1} \tag{6.1}$$

for all k with $|k| \leq t$ and $k \equiv t \pmod{2}$, otherwise 0.

Our lattice path model of this part is equivalent to the lattice paths model of Section 3 with dimension $h = 2$.

Proof of Lemma 6.3. We use the notation of Section 3. Let $h = 2$. We identify a lattice path W_n with a lattice path V of Section 3 in the following way: We identify up-steps with e_1 and down-steps with e_2 . (This corresponds to a rotation of a lattice path W_n by 45 degrees counterclockwise and a scaling by $1/\sqrt{2}$.)

A point $(t, k) \in W_n$ then corresponds to a point in Ω_t , and the uniform distribution follows by Lemma 3.1. \square

Proof of Theorem 6.1. By Lemma 6.3, the expected number of zeros of the path W_n is

$$\mathbb{E}(Z_n) = \sum_{t=0}^n \mathbb{P}((t, 0) \in W_n) = \sum_{t=0}^n [t \text{ even}] \frac{1}{t+1} = \sum_{t=1}^{n+1} \frac{[t \text{ odd}]}{t} = H_{n+1}^{\text{odd}}.$$

\square

Proof of Corollary 6.2. An up-from-zero situation can only occur at a zero of W_n . We have to exclude $(n, 0)$. By Lemma 3.1 going up or down after a zero is equally likely because a zero of W_n corresponds to a diagonal element in a lattice path in the setting of Section 3. Thus

$$\mathbb{E}(Z_n^{\nearrow}) = \frac{1}{2} \left(\mathbb{E}(Z_n) - \frac{[n \text{ even}]}{n+1} \right) = \frac{1}{2} H_n^{\text{odd}}.$$

\square

7. IDENTITY

Using the previous two sections we can show the following identity in a combinatorial way.

Theorem 7.1. *For $n \geq 0$, the following four expressions are equal:*

$$\frac{4}{n+1} \sum_{0 \leq k < \ell < \lceil n/2 \rceil} \frac{\binom{n}{k}}{\binom{n}{\ell}} + [n \text{ even}] \frac{1}{n+1} \left(\frac{2^n}{\binom{n}{n/2}} - 1 \right) + 1 \quad (7.1a)$$

$$= \frac{2}{\lfloor n/2 \rfloor + 1} \sum_{0 \leq k < \ell \leq \lfloor n/2 \rfloor} \frac{\binom{2\lfloor n/2 \rfloor + 1}{k}}{\binom{2\lfloor n/2 \rfloor + 1}{\ell}} + 1 \quad (7.1b)$$

$$= \frac{1}{n+1} \sum_{m=0}^{\lfloor n/2 \rfloor} \sum_{\ell=m}^{n-m} \frac{\binom{2m}{m} \binom{n-2m}{\ell-m}}{\binom{n}{\ell}} \quad (7.1c)$$

$$= H_{n+1}^{\text{odd}}. \quad (7.1d)$$

We note that the expressions (7.1a) and (7.1b) are obviously equal for odd n . Furthermore, (7.1b) and (7.1d) only change when n increases by 2 (to be precise from odd n to even n). Once we prove that (7.1a) equals (7.1d) for all $n \geq 0$, it follows that both expressions are equal to (7.1b) for all $n \geq 0$.

Combinatorial proof of Theorem 7.1. Combine the results of Theorems 5.1 and 6.1 to see that (7.1a) and (7.1d) are equal.

Expression (7.1c) for the expected number of zeros has been given in [2, displayed equation after (14)]: The number of paths from $(0, 0)$ to $(n, n - 2\ell)$ is $\binom{n}{\ell}$, whereas the number of paths from $(0, 0)$ via $(2m, 0)$ to $(n, n - 2\ell)$ is $\binom{2m}{m} \binom{n-2m}{\ell-m}$. Summing over all possible m and all possible ℓ and dividing by $n + 1$ for the equidistribution of the end point yields (7.1c). \square

Beside this combinatorial proof, we intend to show Theorem 7.1 in alternative ways. First, we prove that (7.1c) equals (7.1d): Consider

$$\begin{aligned} \frac{1}{n+1} \sum_{\ell=m}^{n-m} \frac{\binom{2m}{m} \binom{n-2m}{\ell-m}}{\binom{n}{\ell}} &= \frac{1}{n+1} \sum_{\ell=m}^{n-m} \frac{(2m)! (n-2m)! \ell! (n-\ell)!}{m! m! (\ell-m)! (n-\ell-m)! n!} \\ &= \frac{1}{(n+1) \binom{n}{2m}} \sum_{\ell=m}^{n-m} \binom{\ell}{m} \binom{n-\ell}{m} \\ &= \frac{1}{(n+1) \binom{n}{2m}} \binom{n+1}{2m+1} = \frac{1}{2m+1}, \end{aligned}$$

where [4, (5.26)] has been used in the penultimate step. Summing over m yields

$$\sum_{m=0}^{\lfloor n/2 \rfloor} \frac{1}{2m+1} = H_{n+1}^{\text{odd}},$$

thus the equality between (7.1c) and (7.1d).

It remains to give a computational proof of the equality between (7.1a) and (7.1d). We provide two proofs: one motivated by “creative telescoping” (Section 7.1) and one completely elementary “human” proof (Section 7.2) using not more than Vandermonde’s convolution.

7.1. Proof of the Identity Using Creative Telescoping. A computational proof of the identity between (7.1a) and (7.1d) can be generated by the summation package Sigma [14] (see also Schneider [12]) together with the packages HarmonicSums [1] and EvaluateMultiSums [13]³. To succeed, we have to split the case of even and odd n . The obtained proof certificates are rather lengthy to verify.

Motivated by these observations, we also give a proof without computer support. Anyhow, the key step is, as with Sigma, creative telescoping. We prove an (easier) identity, suggested by Sigma, in the following lemma.

Lemma 7.2. *Let*

$$F(n, \ell) = \sum_{0 \leq k < \ell} \frac{\binom{n}{k}}{\binom{n}{\ell}},$$

$$G(n, \ell) = (\ell - 1) + (\ell - 1 - n)F(n, \ell)$$

for $0 \leq \ell \leq n$. Then

$$(n+1)F(n+1, \ell) - (n+2)F(n, \ell) = G(n, \ell+1) - G(n, \ell) \quad (7.2)$$

holds for all $0 \leq \ell < n$.

Proof. For $0 \leq \ell < n$, we first compute

$$\begin{aligned} F(n+1, \ell) &= \frac{1}{\binom{n+1}{\ell}} \sum_{0 \leq k < \ell} \binom{n+1}{k} = \frac{1}{\binom{n+1}{\ell}} \sum_{0 \leq k < \ell} \left(\binom{n}{k-1} + \binom{n}{k} \right) \\ &= -\frac{\binom{n}{\ell-1}}{\binom{n+1}{\ell}} + \frac{2\binom{n}{\ell}}{\binom{n+1}{\ell}} F(n, \ell) = -\frac{\ell}{n+1} + 2\frac{n+1-\ell}{n+1} F(n, \ell) \end{aligned} \quad (7.3)$$

and

$$\begin{aligned} F(n, \ell+1) &= \frac{1}{\binom{n}{\ell+1}} \sum_{0 \leq k < \ell+1} \binom{n}{k} = \frac{\binom{n}{\ell}}{\binom{n}{\ell+1}} + \frac{\binom{n}{\ell}}{\binom{n}{\ell+1}} F(n, \ell) \\ &= \frac{\ell+1}{n-\ell} + \frac{\ell+1}{n-\ell} F(n, \ell). \end{aligned} \quad (7.4)$$

By plugging (7.3) and (7.4) into (7.2), all occurrences of $F(n, \ell)$ cancel as well as all other terms, which proves (7.2). \square

We are now able to prove the essential recurrence for the sum (7.1a) in Theorem 7.1.

Lemma 7.3. *Let*

$$E_n = \frac{4}{n+1} \sum_{0 \leq k < \ell < \lceil n/2 \rceil} \frac{\binom{n}{k}}{\binom{n}{\ell}} + [n \text{ even}] \frac{1}{n+1} \left(\frac{2^n}{\binom{n}{n/2}} - 1 \right).$$

Then

$$E_{2N+1} - E_{2N} = 0 \quad \text{and} \quad E_{2N+2} - E_{2N+1} = \frac{1}{2N+3} \quad (7.5)$$

for $N \geq 0$.

³The authors thank Carsten Schneider for providing the packages Sigma [14] and EvaluateMultiSums [13], and for his support.

Proof. Multiplying (7.2) with $4/((n+1)(n+2))$ and summing over $0 \leq \ell < \lceil n/2 \rceil$ yields

$$\begin{aligned} \frac{4}{n+2} \sum_{0 \leq k < \ell < \lceil n/2 \rceil} \frac{\binom{n+1}{k}}{\binom{n+1}{\ell}} - \frac{4}{n+1} \sum_{0 \leq k < \ell < \lceil n/2 \rceil} \frac{\binom{n}{k}}{\binom{n}{\ell}} \\ = \frac{4}{(n+1)(n+2)} \left(G(n, \lceil n/2 \rceil) - G(n, 0) \right) \end{aligned}$$

for $n \geq 0$. This is equivalent to

$$\begin{aligned} \frac{4}{n+2} \sum_{0 \leq k < \ell < \lceil (n+1)/2 \rceil} \frac{\binom{n+1}{k}}{\binom{n+1}{\ell}} - \frac{4 \lfloor n \text{ even} \rfloor}{n+2} F(n+1, n/2) - \frac{4}{n+1} \sum_{0 \leq k < \ell < \lceil n/2 \rceil} \frac{\binom{n}{k}}{\binom{n}{\ell}} \\ = \frac{4}{(n+1)(n+2)} \left(\lceil n/2 \rceil - 1 + (\lceil n/2 \rceil - 1 - n) F(n, \lceil n/2 \rceil) + 1 \right). \quad (7.6) \end{aligned}$$

We rewrite the double sums in terms of E_n and E_{n+1} , respectively, and use (5.2). We also replace $F(n+1, n/2)$ by (7.3). Then (7.6) is equivalent to

$$\begin{aligned} E_{n+1} - \frac{2 \lfloor n \text{ odd} \rfloor}{n+2} F(n+1, (n+1)/2) - E_n \\ + \frac{2n \lfloor n \text{ even} \rfloor}{(n+1)(n+2)} - \frac{2 \lfloor n \text{ even} \rfloor}{n+1} F(n, n/2) \\ = \frac{4}{(n+1)(n+2)} \left(\lceil n/2 \rceil - (\lfloor n/2 \rfloor + 1) F(n, \lceil n/2 \rceil) \right). \quad (7.7) \end{aligned}$$

If $n = 2N + 1$, equation (7.7) is equivalent to

$$\begin{aligned} E_{2N+2} - E_{2N+1} - \frac{2}{2N+3} F(2N+2, N+1) \\ = \frac{2}{2N+3} - \frac{2}{2N+3} F(2N+1, N+1). \end{aligned}$$

Using (7.3), this is equivalent to the second recurrence in (7.5).

If $n = 2N$, then (7.7) is equivalent to the first recurrence in (7.5). \square

Computational proof of Theorem 7.1. The definition of E_0 in Lemma 7.3 implies that $E_0 = 0$. Thus (7.1a) and (7.1d) coincide for $n = 0$. This can be extended to all $n \geq 0$ by induction on n and Lemma 7.3. \square

7.2. Proof of the Identity Using Vandermonde's Convolution. We now provide a completely elementary “human” proof of the identity between (7.1a) and (7.1d).

We first prove an identity on binomial coefficients.

Lemma 7.4. *The identity*

$$\sum_{0 \leq k \leq j} \binom{n}{k} = \sum_{0 \leq k \leq j} 2^k \binom{n-k-1}{j-k}$$

holds for all non-negative integers $j < n$.

Proof. We denote the right hand side by ρ . The binomial theorem and symmetry of the binomial coefficient yield

$$\rho = \sum_{0 \leq i \leq k \leq j} \binom{k}{i} \binom{n-k-1}{n-1-j}.$$

The sum over k can be evaluated by [4, (5.26)], resulting in

$$\rho = \sum_{0 \leq i \leq j} \binom{n}{n+i-j}.$$

Symmetry of the binomial coefficient and then replacing i by $j-i$ lead to

$$\rho = \sum_{0 \leq i \leq j} \binom{n}{j-i} = \sum_{0 \leq i \leq j} \binom{n}{i}.$$

□

We are now able to establish a recurrence satisfied by the double sum in (7.1a).

Lemma 7.5. *For $n \geq 0$, let*

$$S_n = \sum_{0 \leq k < \ell < \lceil \frac{n}{2} \rceil} \frac{\binom{n}{k}}{\binom{n}{\ell}}.$$

Then the recurrence

$$S_n = \frac{n+1}{n-1} S_{n-2} + \frac{n+1}{4n} + [n \text{ even}] \left(\frac{2^{n-2}}{n \binom{n}{n/2}} - \frac{1}{2(n-1)} - \frac{1}{4n} \right)$$

holds for $n \geq 2$.

Proof. We replace the sum over k in S_n by the term found in Lemma 7.4 and obtain

$$\begin{aligned} S_n &= \sum_{0 \leq k \leq \ell < \lceil \frac{n}{2} \rceil} \left(\frac{\binom{n}{k}}{\binom{n}{\ell}} - \left\lceil \frac{n}{2} \right\rceil \right) \\ &= \sum_{0 \leq k \leq \ell < \lceil \frac{n}{2} \rceil} \frac{2^k \binom{n-k-1}{\ell-k}}{\binom{n}{\ell}} - \left\lceil \frac{n}{2} \right\rceil \\ &= \sum_{0 \leq k \leq \ell < \lceil \frac{n}{2} \rceil} \frac{2^k (n-k-1)! k!}{n!} \binom{\ell}{k} ((n-k) - (\ell-k)) - \left\lceil \frac{n}{2} \right\rceil \\ &= \sum_{0 \leq k \leq \ell < \lceil \frac{n}{2} \rceil} \frac{2^k (n-k)! k!}{n!} \binom{\ell}{k} \\ &\quad - \sum_{0 \leq k \leq \ell < \lceil \frac{n}{2} \rceil} \frac{2^k (n-k-1)! (k+1)!}{n!} \binom{\ell}{k+1} - \left\lceil \frac{n}{2} \right\rceil. \end{aligned}$$

In both sums, the sum over ℓ can be evaluated using upper summation, see [4, Table 174], and we get

$$S_n = \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} \frac{2^k (n-k)! k!}{n!} \binom{\lceil \frac{n}{2} \rceil}{k+1} - \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} \frac{2^k (n-k-1)! (k+1)!}{n!} \binom{\lceil \frac{n}{2} \rceil}{k+2} - \left\lceil \frac{n}{2} \right\rceil.$$

Shifting the summation index in the second sum leads to

$$\begin{aligned} S_n &= \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} \frac{2^k (n-k)! k!}{n!} \binom{\lceil \frac{n}{2} \rceil}{k+1} \\ &\quad - \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} \frac{2^{k-1} (n-k)! k!}{n!} \binom{\lceil \frac{n}{2} \rceil}{k+1} - \left\lceil \frac{n}{2} \right\rceil + \frac{1}{2} \left\lceil \frac{n}{2} \right\rceil \\ &= \frac{1}{2n!} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} 2^k (n-k)! k! \binom{\lceil \frac{n}{2} \rceil}{k+1} - \frac{1}{2} \left\lceil \frac{n}{2} \right\rceil \\ &= \frac{\lceil \frac{n}{2} \rceil!}{2n!} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} 2^k \frac{(n-k)!}{(\lceil \frac{n}{2} \rceil - k - 1)!} \frac{1}{k+1} - \frac{1}{2} \left\lceil \frac{n}{2} \right\rceil. \end{aligned} \quad (7.8)$$

We intend to derive a recurrence linking S_n with S_{n-2} . Therefore, for $n \geq 2$, we rewrite S_n as

$$S_n = \frac{\lceil \frac{n}{2} \rceil!}{2n!} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil - 1} 2^k \frac{(n-k-2)!}{(\lceil \frac{n}{2} \rceil - k - 2)!} \frac{(n-k)(n-k-1)}{(\lceil \frac{n}{2} \rceil - k - 1)(k+1)} - \frac{\lceil \frac{n}{2} \rceil}{2} + \frac{2^{\lceil \frac{n}{2} \rceil - 2}}{\binom{n}{\lceil \frac{n}{2} \rceil - 1}}.$$

Partial fraction decomposition in k yields

$$\begin{aligned} S_n &= \frac{\lceil \frac{n}{2} \rceil!}{2n!} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil - 1} 2^k \frac{(n-k-2)!}{(\lceil \frac{n}{2} \rceil - k - 2)!} \left(-1 + \frac{(n - \lceil \frac{n}{2} \rceil)(n - \lceil \frac{n}{2} \rceil + 1)}{(\lceil \frac{n}{2} \rceil - k - 1)\lceil \frac{n}{2} \rceil} + \frac{n(n+1)}{(k+1)\lceil \frac{n}{2} \rceil} \right) \\ &\quad - \frac{\lceil \frac{n}{2} \rceil}{2} + \frac{2^{\lceil \frac{n}{2} \rceil - 2}}{\binom{n}{\lceil \frac{n}{2} \rceil - 1}} \\ &= -\frac{\lceil \frac{n}{2} \rceil! (n - \lceil \frac{n}{2} \rceil)!}{2n!} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil - 1} 2^k \binom{n-k-2}{\lceil \frac{n}{2} \rceil - k - 2} \\ &\quad + \frac{(\lceil \frac{n}{2} \rceil - 1)!(n - \lceil \frac{n}{2} \rceil + 1)!}{2n!} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil - 1} 2^k \binom{n-k-2}{\lceil \frac{n}{2} \rceil - k - 1} \\ &\quad + \frac{(\lceil \frac{n}{2} \rceil - 1)!(n+1)n}{2n!} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil - 1} 2^k \frac{(n-k-2)!}{(\lceil \frac{n}{2} \rceil - k - 2)!} \frac{1}{(k+1)} \\ &\quad - \frac{\lceil \frac{n}{2} \rceil}{2} + \frac{2^{\lceil \frac{n}{2} \rceil - 2}}{\binom{n}{\lceil \frac{n}{2} \rceil - 1}}. \end{aligned}$$

We again use Lemma 7.4 for the first two summands and (7.8) with n replaced by $n - 2$ for the third summand to obtain

$$\begin{aligned} S_n &= -\frac{1}{2\binom{n}{\lceil \frac{n}{2} \rceil}} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil - 1} \binom{n-1}{k} + \frac{1}{2\binom{n}{\lceil \frac{n}{2} \rceil - 1}} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} \binom{n-1}{k} \\ &\quad + \frac{n+1}{n-1} \left(S_{n-2} + \frac{\lceil \frac{n}{2} \rceil - 1}{2} \right) - \frac{\lceil \frac{n}{2} \rceil}{2}. \end{aligned}$$

Adding another summand for $k = \lceil \frac{n}{2} \rceil - 1$ to the first sum leads to

$$\begin{aligned} S_n &= \left(\frac{1}{2\binom{n}{\lceil \frac{n}{2} \rceil - 1}} - \frac{1}{2\binom{n}{\lceil \frac{n}{2} \rceil}} \right) \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} \binom{n-1}{k} \\ &\quad + \frac{\binom{n-1}{\lceil \frac{n}{2} \rceil - 1}}{2\binom{n}{\lceil \frac{n}{2} \rceil}} + \frac{n+1}{n-1} S_{n-2} + \frac{\lceil \frac{n}{2} \rceil - 1}{n-1} - \frac{1}{2} \\ &= \frac{n - 2\lceil \frac{n}{2} \rceil + 1}{2\lceil \frac{n}{2} \rceil \binom{n}{\lceil \frac{n}{2} \rceil}} \sum_{0 \leq k < \lceil \frac{n}{2} \rceil} \binom{n-1}{k} \\ &\quad + \frac{n+1}{n-1} S_{n-2} + \frac{\lceil \frac{n}{2} \rceil}{2n} + \frac{\lceil \frac{n}{2} \rceil - 1}{n-1} - \frac{1}{2}. \end{aligned}$$

If n is odd, then $n - 2\lceil \frac{n}{2} \rceil + 1 = 0$ so that the first summand vanishes. The result follows in that case.

For even n , the remaining sum is 2^{n-2} and the result follows. \square

Computational proof of Theorem 7.1. Denote the expression in (7.1a) by E_n . From Lemma 7.5, we obtain the recurrence

$$E_n = E_{n-2} + \frac{1}{n + [n \text{ even}]}$$

for $n \geq 2$. As $E_0 = E_1 = 1$, this implies that $E_n = H_{n+1}^{\text{odd}}$. Thus (7.1a) and (7.1d) coincide. \square

8. ASYMPTOTIC ASPECTS

Lemma 8.1. *We have*

$$H_n = \log n + \gamma + \frac{1}{2n} - \frac{1}{12n^2} + O\left(\frac{1}{n^4}\right), \quad (8.1a)$$

$$H_n^{\text{odd}} = \frac{1}{2} \log n + \frac{\gamma + \log 2}{2} + \frac{[n \text{ odd}]}{2n} + \frac{3[n \text{ even}] - 2}{12n^2} + O\left(\frac{1}{n^4}\right), \quad (8.1b)$$

$$H_n^{\text{alt}} = -\log 2 + \frac{(-1)^n}{2n} - \frac{(-1)^n}{4n^2} + O\left(\frac{1}{n^4}\right) \quad (8.1c)$$

as n tends to ∞ , and where $\gamma = 0.5772156649\dots$ is the Euler–Mascheroni constant.

Proof. The asymptotic expansion (8.1a) is well-known, cf. for instance [4, (9.88)].

We write $\alpha_n = [n \text{ even}]$ and thus $[n/2] = (n - 1 + \alpha_n)/2$. As α_n is obviously bounded, we can simply plugin this expression into the asymptotic expansions and simplify all occurring higher powers of α_n by the fact that $\alpha_n^2 = \alpha_n$. Using the relations $H_n^{\text{odd}} = H_n - H_{[n/2]}/2$ and $H_n^{\text{alt}} = H_n - 2H_n^{\text{odd}}$ leads to the expansions (8.1b) and (8.1c), respectively.

The actual asymptotic computations⁴ have been carried out using the asymptotic expansions module [5] of SageMath [11]. \square

Corollary 8.2. *The expected numbers of zeros and up-from-zero situations are*

$$\begin{aligned}\mathbb{E}(Z_n) &= \frac{1}{2} \log n + \frac{\gamma + \log 2}{2} + \frac{1 + [n \text{ even}]}{2n} - \frac{2 + 9[n \text{ even}]}{12n^2} + \frac{[n \text{ even}]}{n^3} + O\left(\frac{1}{n^4}\right), \\ \mathbb{E}(Z_n^\nearrow) &= \frac{1}{4} \log n + \frac{\gamma + \log 2}{4} + \frac{[n \text{ odd}]}{4n} + \frac{3[n \text{ even}] - 2}{24n^2} + O\left(\frac{1}{n^4}\right),\end{aligned}$$

respectively, asymptotically as n tends to infinity.

Proof. Combine Theorem 6.1 and Corollary 6.2 with Lemma 8.1. \square

9. DISTRIBUTION

In this section, we study the distribution of the number of zeros. As for the expectation $\mathbb{E}(Z_n)$, we get an exact formula as well as an asymptotic formula.

Theorem 9.1. *Let $r \in \mathbb{N}_0$. For positive lengths $n \geq 2r - 2$, the probability that a randomly chosen path W_n has exactly r zeros is*

$$\begin{aligned}\mathbb{P}(Z_n = r) &= \frac{2^r}{n+1} \frac{\binom{[n/2]}{r}}{\binom{n}{r}} \left(\frac{2[n/2]}{r(r+1)} + \frac{r-1}{r+1} + [n \text{ even}] \frac{1}{r} \right) \\ &\quad + [n \text{ even}] \frac{2^{r-1}(r-1)}{(n+1)r} \frac{\binom{n/2}{r-1}}{\binom{n}{r}}\end{aligned}$$

and we have $\mathbb{P}(Z_0 = r) = [r = 1]$. Otherwise $\mathbb{P}(Z_n = r) = 0$.

This exact formula admits a local limit theorem towards a discrete distribution. The details are as follows.

Corollary 9.2. *Let $0 < \varepsilon \leq \frac{1}{2}$. For positive integers r with $r = O(n^{1/2-\varepsilon})$, we have asymptotically*

$$\mathbb{P}(Z_n = r) = \frac{1}{r(r+1)} (1 + O(1/n^{2\varepsilon}))$$

as n tends to infinity.

Proof of Theorem 9.1. Again, we assume $d \geq 0$ (by symmetry of the generating function (5.1)). Note that Q_d counts the number of zeros by the variable u except for the first zero (at $(0,0)$). By starting with Lemma 5.2 and some rewriting, we can extract the $(r-1)$ st coefficient with respect to u as

$$[u^{r-1}]Q_d(z, u) = [u^{r-1}] \frac{v^d}{1 - u \frac{2v^2}{1+v^2}} = \frac{2^{r-1}v^{2(r-1)+d}}{(1+v^2)^{r-1}}.$$

Next, we extract the coefficient of z^n . We use Lemma 5.3 to obtain

$$\begin{aligned}[z^n u^{r-1}]Q_d(z, u) &= [v^n](1-v^2)(1+v^2)^{n-1} \frac{2^{r-1}v^{2(r-1)+d}}{(1+v^2)^{r-1}} \\ &= 2^{r-1} [v^{n-d-2(r-1)}](1-v^2)(1+v^2)^{n-r}.\end{aligned}$$

⁴A worksheet containing the computations can be found at <http://www.danielkrenn.at/downloads/quicksort-paths-full/quicksort-paths.ipynb>.

We set $\ell = \frac{1}{2}(n-d)$ and get

$$\begin{aligned} [z^n u^{r-1}]Q_d(z, u) &= 2^{r-1} [v^{2\ell-2r+2}](1-v^2)(1+v^2)^{n-r} \\ &= 2^{r-1} [v^{\ell-r+1}](1-v)(1+v)^{n-r} \\ &= 2^{r-1} \binom{n-r}{\ell-r+1} - 2^{r-1} \binom{n-r}{\ell-r} \\ &= 2^{r-1} \binom{n-r}{n-\ell-1} - 2^{r-1} \binom{n-r}{n-\ell}. \end{aligned}$$

Note that we have to assume $n-r \geq 0$ to make this work. Otherwise, anyhow, there are no paths with exactly r zeros (and positive length n).

If $\ell > r-1$ we can rewrite the previous formula to obtain

$$[z^n u^{r-1}]Q_d(z, u) = 2^{r-1} \binom{n-r}{n-\ell} \left(\frac{n-\ell}{\ell-r+1} - 1 \right) = 2^{r-1} \frac{d+r-1}{\ell-r+1} \binom{n-r}{n-\ell},$$

if $\ell = r-1$, then we have $[z^n u^{r-1}]Q_d(z, u) = 2^{r-1}$ (independently of n), and if $\ell < r-1$ we get $[z^n u^{r-1}]Q_d(z, u) = 0$.

To finish the proof, we have to normalize this number of paths with exactly r zeros and then sum up over all ℓ . So let us start with the normalization part. We set

$$\lambda_{n,r,d} = \mathbb{P}(Z_n = r \mid W_n \text{ ends in } (n, d)) = \frac{[z^n u^{r-1}]Q_d(z, u)}{[z^n]Q_d(z, 1)}$$

for $n \equiv d \pmod{2}$ and $\lambda_{n,r,d} = 0$ otherwise. The denominator $[z^n]Q_d(z, 1)$ was already determined in Lemma 5.4.

If $\ell > r-1$, we have

$$\begin{aligned} \lambda_{n,r,d} &= 2^{r-1} \frac{d+r-1}{\ell-r+1} \binom{n-r}{n-\ell} / \binom{n}{\ell} \\ &= 2^{r-1} \frac{d+r-1}{\ell-r+1} \frac{(n-r)! \ell! (n-\ell)!}{(n-\ell)! (\ell-r)! n!} \\ &= 2^{r-1} \frac{(n-r)! \ell! (d+r-1)}{n! (\ell-r+1)!}, \end{aligned}$$

where the last line magically holds for $\ell = r-1$ as well. In particular, we obtain

$$\lambda_{n,r,0} = 2^{r-1} (r-1) [n \geq 2r-2] \frac{(n/2)!}{n!} \frac{(n-r)!}{(n/2-r+1)!} = \frac{2^{r-1} (r-1)}{r} \binom{n/2}{r-1} / \binom{n}{r}.$$

We have arrived at the summation of the $\lambda_{n,r,d}$. The result follows as

$$\begin{aligned} \mathbb{P}(Z_n = r) &= \sum_{d=-n}^n \mathbb{P}(Z_n = r \mid W_n \text{ ends in } (n, d)) \mathbb{P}(W_n \text{ ends in } (n, d)) \\ &= \frac{1}{n+1} \sum_{d=-n}^n \lambda_{n,r,d} \\ &= \frac{2}{n+1} \sum_{\ell=0}^{\lceil n/2 \rceil - 1} \lambda_{n,r,n-2\ell} + \frac{1}{n+1} \lambda_{n,r,0}, \end{aligned}$$

and plugging in $\lambda_{n,r,d}$ gives the intermediate result

$$\begin{aligned} \mathbb{P}(Z_n = r) &= 2^r \frac{(n-r)!}{(n+1)!} \sum_{\ell=r-1}^{\lceil n/2 \rceil - 1} \frac{\ell! (n-2\ell+r-1)}{(\ell-r+1)!} \\ &\quad + [n \text{ even}] \frac{2^{r-1}(r-1)}{(n+1)r} \binom{n/2}{r-1} / \binom{n}{r} \end{aligned} \quad (9.1)$$

for $n \geq r$. At this point, we interrupt this proof to evaluate the sum over the ℓ and continue afterwards. \square

Lemma 9.3. *We have*

$$\sum_{\ell=r-1}^{\lceil n/2 \rceil - 1} \frac{\ell! (n-2\ell+r-1)}{(\ell-r+1)!} = r! \binom{\lceil n/2 \rceil}{r} \left(\frac{2^{\lceil \frac{n}{2} \rceil}}{r(r+1)} + \frac{r-1}{r+1} + [n \text{ even}] \frac{1}{r} \right).$$

Of course, this lemma can be proved computationally by, for example, Sigma [14]. However, we give a direct proof here.

Proof. We obtain

$$\begin{aligned} \sum_{\ell=r-1}^{\lceil n/2 \rceil - 1} \frac{\ell! (n-2\ell+r-1)}{(\ell-r+1)!} &= (r-1)! \sum_{\ell=r-1}^{\lceil n/2 \rceil - 1} \binom{\ell}{r-1} (n-2\ell+r-1) \\ &= (r-1)! \sum_{\ell=r-1}^{\lceil n/2 \rceil - 1} \binom{\ell}{r-1} (n+1+r-2(\ell+1)) \\ &= (n+1+r)(r-1)! \sum_{\ell=r-1}^{\lceil n/2 \rceil - 1} \binom{\ell}{r-1} - 2r! \sum_{\ell=r-1}^{\lceil n/2 \rceil - 1} \binom{\ell+1}{r}. \end{aligned}$$

Using upper summation, cf. [4, 5.10], yields

$$\begin{aligned} \sum_{\ell=r-1}^{\lceil n/2 \rceil - 1} \frac{\ell! (n-2\ell+r-1)}{(\ell-r+1)!} &= (n+1+r)(r-1)! \left(\binom{\lceil n/2 \rceil}{r} - 2r! \binom{\lceil n/2 \rceil + 1}{r+1} \right) \\ &= (r-1)! \binom{\lceil n/2 \rceil}{r} \left(n+1+r - 2 \frac{r}{r+1} (\lceil n/2 \rceil + 1) \right) \\ &= (r-1)! \binom{\lceil n/2 \rceil}{r} \left(n+r-1 - 2 \lceil n/2 \rceil + \frac{2 \lceil n/2 \rceil}{r+1} + \frac{2}{r+1} \right). \end{aligned}$$

The lemma follows by replacing $n-2 \lceil n/2 \rceil$ with $[n \text{ even}] - 1$ and by collecting terms. \square

Proof of Theorem 9.1 continued. To finish the proof, we simply plug in the result of Lemma 9.3 into (9.1) and rewrite the factorials as binomial coefficients. \square

As a next step, we want to prove Corollary 9.2, which extracts the asymptotic behavior of the distribution (Theorem 9.1). To show that asymptotic formula, we will use the following auxiliary result.

Lemma 9.4. *Let $0 < \varepsilon \leq \frac{1}{2}$. For integers c with $c = O(N^{1/2-\varepsilon})$ we have*

$$c! \binom{N}{c} = N^c (1 + O(1/N^{2\varepsilon})).$$

Proof. The inequality $N^c \geq c! \binom{N}{c}$ is trivial. We observe

$$\begin{aligned} c! \binom{N}{c} &= N^c \cdot \prod_{0 \leq i < c} \left(1 - \frac{i}{N}\right) \geq N^c \cdot \left(1 - \sum_{0 \leq i < c} \frac{i}{N}\right) \geq N^c \left(1 - \frac{c^2}{2N}\right) \\ &= N^c (1 + O(1/N^{2\varepsilon})), \end{aligned}$$

where the assumption on c has been used in the last step. \square

Proof of Corollary 9.2. By using Lemma 9.4, the exact result of Theorem 9.1 becomes

$$\begin{aligned} \mathbb{P}(Z_n = r) &= \frac{2^r}{n} \frac{n^r}{2^r} \frac{1}{n^r} \left(\frac{n}{r(r+1)} + O(1) \right) (1 + O(1/n^{2\varepsilon})) \\ &\quad + [n \text{ even}] \frac{2^{r-1}(r-1)}{n} \frac{n^{r-1}}{2^{r-1}} \frac{1}{n^r} (1 + O(1/n^{2\varepsilon})) \\ &= \frac{1}{r(r+1)} (1 + O(1/n^{2\varepsilon})) \end{aligned}$$

as claimed. \square

Part II. Dual-Pivot Quicksort

In this second part of this work, we analyze a partitioning strategy and the dual-pivot quicksort algorithm itself.

As mentioned in the introduction, the number of key comparisons of dual-pivot quicksort depends on the concrete partitioning procedure. This is in contrast to the standard quicksort algorithm with only one pivot, where partitioning always has cost $n - 1$ (for partitioning n elements; one is taken as the pivot). For example, if one wants to classify a large element, i. e., an element larger than the larger pivot, comparing it with the larger pivot is unavoidable, but it depends on the partitioning procedure whether a comparison with the smaller pivot occurs as well.

First, in Section 10, we make the set-up precise, fix notions, and start solving the dual-pivot quicksort recurrence (10.6). This recurrence relates the cost of the partitioning step to the total number of comparisons of dual-pivot quicksort. These results are independent of the partition strategy.

In Section 11 the partitioning strategy “Count” is introduced and its cost is analyzed. It will turn out that the results on lattice paths obtained in Part I are central in determining the partitioning cost.

Everything is put together in Section 12: We obtain the exact comparison count (Theorem 12.1). The asymptotic behavior is extracted out of the exact results (Corollary 12.2). In Appendix C we report on experiments that validate these results for small input sizes and show that empirical measurements over only a few inputs come close to these exact results for large input sizes.

10. SOLUTION OF THE DUAL-PIVOT QUICKSORT RECURRENCE

We consider versions of dual-pivot quicksort that act as follows on an input sequence (a_1, \dots, a_n) consisting of distinct numbers: If $n \leq 1$, do nothing, otherwise choose a_1 and a_n as pivots, and by one comparison determine $p = \min(a_1, a_n)$ and $q = \max(a_1, a_n)$. Use a partitioning procedure to partition the remaining $n - 2$ elements into the three classes *small*, *medium*, and *large*. Then call dual-pivot

quicksort recursively on each of these three classes to finish the sorting, using the same partitioning procedure in all recursive calls.

We denote the number of small, medium and large elements defined by the pivot elements by S , M and L respectively.

Of course, the random variables S , M and L depend on n . For simplicity and readability, this is not reflected in the notation. However, our discussion heavily uses generating functions, where we need to make the dependence on n explicit; so we will write S_n , M_n and L_n in this context.

We will need the following results on the distribution of (S, M, L) .

Lemma 10.1. *The triple (S, M, L) is uniformly distributed on*

$$\Omega' := \{(s, m, \ell) \mid s + m + \ell = n - 2, s, m, \ell \in \mathbb{N}_0\}. \quad (10.1)$$

The random variables S , M and L are identically distributed and we have

$$\mathbb{P}(M = m) = \frac{n - m - 1}{\binom{n}{2}} \quad \text{for } 0 \leq m \leq n - 2, \quad (10.2)$$

$$\mathbb{E}(M) = \frac{n - 2}{3}, \quad (10.3)$$

$$\sum_{n \geq 2} \mathbb{E}(|L_n - S_n|) z^n = \frac{1}{3(1 - z)^2} - \frac{1}{2(1 - z)} - \frac{1}{2}(z + 1) \operatorname{artanh}(z) + \frac{1}{6} + \frac{1}{3}z. \quad (10.4)$$

Proof. Each pair of pivot elements is equally likely and there is an obvious bijection between the pairs of pivot elements and the triples $(S, M, L) \in \Omega'$. Thus (S, M, L) is uniformly distributed on Ω' . This (or direct enumeration of Ω') implies that that Ω' has cardinality $\binom{n}{2}$.

By the symmetry in the definition of Ω' , it is clear that S , M and L are identically distributed. Thus $3\mathbb{E}(M) = \mathbb{E}(S) + \mathbb{E}(M) + \mathbb{E}(L) = \mathbb{E}(n - 2) = n - 2$, which implies (10.3). For $0 \leq m \leq n - 2$, there are $n - m - 1$ choices for (s, ℓ) such that $(s, m, \ell) \in \Omega'$. Together with $|\Omega'| = \binom{n}{2}$, this implies (10.2).

Let $g_{d,n}$ be the number of triples $(s, m, \ell) \in \Omega'$ with $|\ell - s| = d$. We can write the bivariate generating function of $g_{d,n}$ as

$$G(u, z) := \sum_{\substack{n \geq 0 \\ d \geq 0}} g_{d,n} u^d z^n = \frac{1}{1 - z^2} \frac{1}{1 - z} \left(1 + \frac{2uz}{1 - uz}\right).$$

To see this, first only consider the cases that $\ell > s$. Write $s + m + \ell = s + m + (s + (\ell - s)) = 2s + m + (\ell - s)$. As $\ell - s$ is marked by the variable u , this triple (s, m, ℓ) contributes $(z^2)^s z^m (zu)^{\ell - s}$. Summing over all these triples yields $1/(1 - z^2) \cdot 1/(1 - z) \cdot uz/(1 - uz)$. The triples with $\ell < s$ contribute the same, thus the factor 2 in the result. The triples with $\ell = s$ contribute $1/(1 - z^2) \cdot 1/(1 - z)$.

Therefore,

$$\mathbb{E}(|L_n - S_n|) = \frac{1}{\binom{n}{2}} [z^{n-2}] \frac{\partial G(u, z)}{\partial u} \Big|_{u=1} = \frac{2}{n(n-1)} [z^{n-2}] \frac{2z}{(1+z)(1-z^4)}. \quad (10.5)$$

We have to compute the generating function

$$H(z) := \sum_{n \geq 2} \mathbb{E}(|L_n - S_n|) z^n.$$

Differentiating twice yields and using (10.5) yields

$$H''(z) = \sum_{n \geq 2} n(n-1) \mathbb{E}(|L_n - S_n|) z^{n-2} = \frac{4z}{(1+z)(1-z^4)}.$$

Integrating twice yields (10.4). \square

Let P_n , a random variable, denote the *partitioning cost*. This is defined as the number of comparisons made by the partitioning procedure if the input (a_1, \dots, a_n) is assumed to be in random order. Further, let C_n be the random variable that denotes the number of comparisons carried out when sorting n elements with dual-pivot quicksort. The reader should be aware that both P_n and C_n are determined by the partitioning procedure used.

Since the input (a_1, \dots, a_n) is in random order and the partitioning procedure does nothing but compare elements with the two pivots, the inputs for the recursive calls are in random order as well, which implies that the distributions of P_n and C_n only depend on n .

The recurrence

$$\mathbb{E}(C_n) = \mathbb{E}(P_n) + \frac{3}{\binom{n}{2}} \sum_{k=1}^{n-2} (n-1-k) \mathbb{E}(C_k) \quad (10.6)$$

for $n \geq 0$ describes the connection between the expected sorting cost $\mathbb{E}(C_n)$ and the expected partitioning cost $\mathbb{E}(P_n)$. It will be central for our analysis. Note that it is irrelevant for (10.6) how the partitioning cost $\mathbb{E}(P_n)$ is determined; it need not even be referring to comparisons. The recurrence is simple and well-known; a version of it occurs already in Sedgewick's thesis [15]. For the convenience of the reader we give a brief proof: We clearly have $C_n = P_n + C_S + C_M + C_L$. Taking expectations and using the fact that S, M, L and therefore C_S, C_M, C_L are identically distributed as well as the law of total expectation yields

$$\begin{aligned} \mathbb{E}(C_n) &= \mathbb{E}(P_n) + \mathbb{E}(C_S) + \mathbb{E}(C_M) + \mathbb{E}(C_L) = \mathbb{E}(P_n) + 3 \mathbb{E}(C_M) \\ &= \mathbb{E}(P_n) + 3 \sum_{k=0}^{n-2} \mathbb{P}(M = k) \mathbb{E}(C_k). \end{aligned}$$

Using (10.2) leads to (10.6).

We recall how to solve recurrence (10.6) using generating functions. We follow Wild [16, § 4.2.2] who in turn follows Hennequin [6]. The following lemma is contained in slightly different notation in [16].

Lemma 10.2. *With the cost C_n and P_n as above, $C(z) = \sum_{n \geq 0} \mathbb{E}(C_n) z^n$ and $P(z) = \sum_{n \geq 0} \mathbb{E}(P_n) z^n$, we have*

$$C(z) = (1-z)^3 \int_0^z (1-t)^{-6} \int_0^t (1-s)^3 P''(s) ds dt.$$

For self-containedness, the proof is in Appendix A.

11. PARTITIONING COST

In Section 10 we saw that in order to calculate the average number of comparisons $\mathbb{E}(C_n)$ of a dual-pivot quicksort algorithm we need the expected partitioning cost $\mathbb{E}(P_n)$ of the partitioning procedure used. The aim of this section is to determine $\mathbb{E}(P_n)$ for the partitioning procedure “Count” to be described below.

We use the set-up described at the beginning of Section 10. For partitioning we use comparisons to *classify* the $n - 2$ elements a_2, \dots, a_{n-1} as *small*, *medium*, or *large*. We will be using the term *classification* for this central aspect of partitioning. Details of a partitioning procedure that concern how the classes are represented or elements are moved around may and will be ignored. (Nonetheless, in Appendix D we provide pseudocode for the considered classification strategies turned into dual-pivot quicksort algorithms.) The cost P_n depends on the concrete classification strategy used, the only relevant difference between classification strategies being whether the next element to be classified is compared with the smaller pivot p or the larger pivot q first. This decision may depend on the whole history of outcomes of previous comparisons. (The resulting abstract classification strategies may conveniently be described as classification trees, see [2], but we do not need this model here. See also Section 14.)

Two comparisons are necessary for each medium element. Furthermore, one comparison with p is necessary for small and one comparison with q is necessary for large elements. We call other comparisons occurring during classification *additional comparisons*. That means, an additional comparison arises when a small element is compared with q first or a large element is compared with p first.

Next we describe the classification strategy “Count” from [2]. Let s_t and ℓ_t denote the number of elements that have been classified as small and large, respectively, in the first t classification rounds. Set $s_0 = \ell_0 = 0$.

Strategy “Count”. *When classifying the t -th element, for $1 \leq t \leq n - 2$, proceed as follows: If $s_{t-1} \geq \ell_{t-1}$, compare with p first, otherwise compare with q first.*

In Appendix B we describe a closely related strategy named “Clairvoyant”, which uses an oracle and uses slightly fewer comparisons than “Count”. In [2], “Clairvoyant” was used as an auxiliary means for analyzing “Count”. In Part III, we show that “Count” is an optimal algorithmic strategy.

The goal of this section is to prove the following proposition.

Proposition 11.1. *Let $n \geq 2$. The expected partitioning cost of strategy “Count” is*

$$\mathbb{E}(P_n) = \frac{3}{2}n + \frac{1}{2}H_n^{\text{odd}} - \frac{19}{8} - \frac{3[n \text{ odd}]}{8n} - \frac{[n \text{ even}]}{8(n-1)}. \quad (11.1)$$

The corresponding generating function is

$$\begin{aligned} P(z) = \sum_{n \geq 2} \mathbb{E}(P_n) z^n &= \frac{3}{2(1-z)^2} + \frac{\text{artanh}(z)}{2(1-z)} \\ &\quad - \frac{31z^2}{8(1-z)} - \frac{3+z}{8} \text{artanh}(z) - \frac{3}{2} - \frac{25z}{8}. \end{aligned} \quad (11.2)$$

Asymptotically, we have

$$\mathbb{E}(P_n) = \frac{3}{2}n + \frac{1}{4} \log n + \frac{1}{4} \gamma + \frac{\log 2}{4} - \frac{19}{8} - \frac{1}{8n} - \frac{1}{12n^2} + O\left(\frac{1}{n^3}\right)$$

for $n \rightarrow \infty$.

We now translate the elements of the random permutation (a_1, \dots, a_n) that we are reading into a random walk W starting at the origin and using *up-steps* $(1, +1)$ when a large element is encountered, *right-steps* $(1, 0)$ when a medium element is

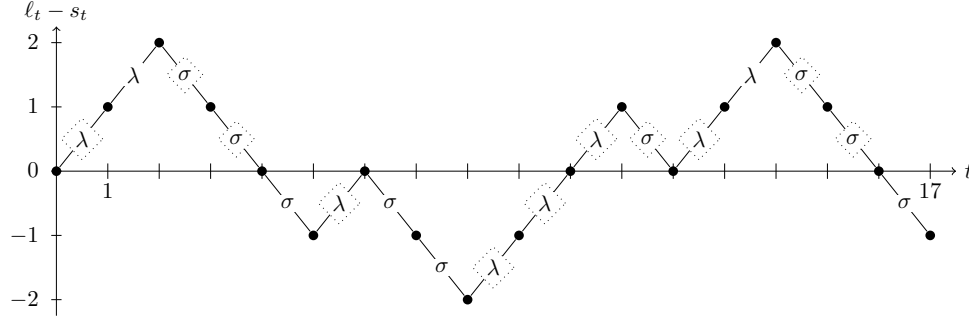


FIGURE 11.1. Example run of strategy “Count” on input $b = (\lambda, \lambda, \sigma, \sigma, \sigma, \lambda, \sigma, \sigma, \lambda, \lambda, \lambda, \sigma, \lambda, \lambda, \sigma, \sigma, \sigma)$. The $s = 9$ small elements in b are represented by the symbol σ , the $\ell = 8$ large elements by λ (no medium elements μ present). The horizontal axis shows the input position t , the vertical axis shows the difference $\ell_t - s_t$ of large and small elements in the initial segment $b_{\leq t} = (b_1, \dots, b_t)$ of the input. A diamond marks elements where an additional comparison occurs using strategy “Count”. It makes eleven additional comparisons on this input.

encountered and *down-steps* $(1, -1)$ when a small element is encountered. Removing all M right-steps from W leads to a walk W' of length $n - 2 - M$ which only has up-steps and down-steps, see Figure 11.1.

It now turns out that W' is a path which is distributed exactly as the random paths studied in Part I.

Lemma 11.2. *Let $0 \leq m \leq n - 2$. We have*

$$\mathbb{P}(W' \text{ ends at } (n - m - 2, d) \mid M = m) = \frac{1}{n - m - 1} \quad (11.3)$$

for all $d \equiv n - m - 2 \pmod{2}$ and $|d| \leq n - m - 2$.

For such m and d , all walks W' ending in $(n - m - 2, d)$ are equally likely.

Proof. As there are L up-steps and S down-steps, the walk W ends at $(n - 2, L - S)$. Thus W' ends on $(n - 2 - M, L - S)$.

Once $S = s$, $M = m$ and $L = \ell$ are fixed, there are $\binom{n-2}{s, m, \ell}$ walks with that numbers of up-steps, right-steps and down-steps, respectively. All of them are equally likely because they only differ by the order of the steps and all orders are equally likely by our probability model on the input list. Thus,

$$\mathbb{P}(W = w \mid (S, M, L) = (s, m, \ell)) = \frac{1}{\binom{n-2}{s, m, \ell}}$$

for all walks w with ℓ up-steps, m right-steps and s down-steps.

We have

$$\mathbb{P}(W' \text{ ends at } (n - m - 2, d) \mid M = m) = \frac{\mathbb{P}(L - S = d \text{ and } L + S = n - m - 2)}{\mathbb{P}(M = m)}.$$

The expression in the numerator uniquely determines the pair (S, L) . As each of these pairs with $S + m + L = n - 2$ is equally likely, the numerator cannot depend on d if it is positive at all. It is easily seen that the probability is positive if and

only if $d \equiv n - m - 2 \pmod{2}$ and $|d| \leq n - m - 2$. There are $n - m - 1$ choices for d , thus we obtain (11.3).

Each walk w' of length $n - m - 2$ corresponds to $\binom{n-2}{m}$ walks w with up-, right- and down-steps of length $n - 2$. Thus all walks w' ending at $(n - m - 2, d)$ are equally likely. \square

We are now able to describe the partitioning cost.

Lemma 11.3. *For $n \geq 2$, the partitioning cost P_n is*

$$P_n = 1 + \frac{3}{2}(n - 2) + \frac{1}{2}M + Z_{n-2-M}^{\nearrow} - \frac{1}{2}|L - S| \quad (11.4)$$

where Z_{n-2-M}^{\nearrow} denotes the number of up-from-zero situations (cf. Section 6) of the random walk W' .

Proof. The first summand 1 corresponds to the comparison between the two pivot elements. We note that the number of up-from-zero situations of W equals Z_{n-2-M}^{\nearrow} by the construction of W' , because omitting the right-steps clearly does not change the number of up-from-zero situations.

A right-step (medium element) always has partitioning cost of $2 = \frac{3}{2} + \frac{1}{2}$. We split the cost of 2 induced by an up-from-zero situation (compare with smaller pivot first, but read a large element) into $1 + 1$, the latter being taken into account by the summand Z_{n-2-M}^{\nearrow} . Thus, for the remainder of this proof, we only have to consider a remaining cost of 1 for those steps. A step away from the axis (correct pivot first) then costs $1 = 3/2 - 1/2$, a step towards the axis (wrong pivot first) costs $2 = 3/2 + 1/2$. The sum of the numbers of these steps is $n - 2 - M$, their difference is $|L - S|$. This amounts to a contribution of $\frac{3}{2}(n - M - 2) - \frac{1}{2}|L - S|$. \square

In order to compute the generating function of $\mathbb{E}(P_n)$, we have to compute the generating function of $\mathbb{E}(Z_{n-2-M_n}^{\nearrow})$.

Lemma 11.4. *The generating function of $\mathbb{E}(Z_{n-2-M_n}^{\nearrow})$ is*

$$\sum_{n \geq 2} \mathbb{E}(Z_{n-2-M_n}^{\nearrow}) z^n = \frac{\operatorname{artanh}(z)}{2(1-z)} - \frac{z^2}{8(1-z)} - \frac{3z+5}{8} \operatorname{artanh}(z) + \frac{1}{8}z. \quad (11.5)$$

Proof. The law of total expectation yields

$$\mathbb{E}(Z_{n-2-M_n}^{\nearrow}) = \sum_{m=0}^{n-2} \mathbb{P}(M_n = m) \mathbb{E}(Z_{n-2-m}^{\nearrow}).$$

From (10.2) and Corollary 6.2, we immediately get that

$$\begin{aligned} \mathbb{E}(Z_{n-2-M_n}^{\nearrow}) &= \frac{1}{2\binom{n}{2}} \sum_{m=0}^{n-2} (n - m - 1) H_{n-2-m}^{\text{odd}} = \frac{1}{2\binom{n}{2}} \sum_{m=0}^{n-2} (m + 1) H_m^{\text{odd}} \\ &= \frac{1}{2\binom{n}{2}} \sum_{m=0}^{n-2} \sum_{k=1}^m [k \text{ odd}] \frac{m+1}{k}. \end{aligned} \quad (11.6)$$

We now consider the generating function

$$G(z) := \sum_{n \geq 2} \mathbb{E}(Z_{n-2-M_n}^{\nearrow}) z^n.$$

By (11.6), we have

$$G'''(z) = \sum_{n \geq 2} \sum_{m=0}^{n-2} \sum_{k=1}^m \frac{[k \text{ odd}]}{k} (m+1) z^{n-2}.$$

Exchanging the order of summation and replacing $n-2$ by n yields

$$\begin{aligned} G''(z) &= \sum_{k \geq 1} \frac{[k \text{ odd}]}{k} \sum_{m \geq k} (m+1) \sum_{n \geq m} z^n \\ &= \frac{1}{1-z} \sum_{k \geq 1} \frac{[k \text{ odd}]}{k} \sum_{m \geq k} (m+1) z^m \\ &= \frac{1}{1-z} \sum_{k \geq 1} \frac{[k \text{ odd}]}{k} \left(\frac{kz^k}{1-z} + \frac{z^k}{(1-z)^2} \right) \\ &= \frac{1}{(1-z)^2} \sum_{k \geq 1} [k \text{ odd}] z^k + \frac{1}{(1-z)^3} \sum_{k \geq 1} \frac{[k \text{ odd}]}{k} z^k \\ &= \frac{z}{(1-z)^3(1+z)} + \frac{1}{(1-z)^3} \operatorname{artanh}(z). \end{aligned}$$

Integrating twice yields (11.5). Note that the summand $1/8z$ ensures that $G(z) = O(z^2)$. \square

Proof of Proposition 11.1. Taking expectations in (11.4) and summing up the contributions of (10.3), (10.4) and Lemma 11.4 yields (11.2).

For deriving (11.1), we use the expansions

$$\sum_{n > k} \frac{[n-k \text{ odd}]}{n-k} z^n = z^k \operatorname{artanh}(z). \quad (11.7)$$

valid for $k \in \mathbb{Z}$. The asymptotic expansion follows from Lemma 8.1. \square

12. MAIN RESULTS AND THEIR ASYMPTOTIC ASPECTS

In this section we give precise formulations and proofs of our main results. We use the partitioning cost from the previous section to calculate the expected number of key comparisons of the dual-pivot quicksort variant obtained by using classification strategy “Count”. We call this sorting algorithm “Count” again. We validated our main results in experiments which can be found in Appendix C. They show that the error term $O(n^{-4})$ is small already for real-life input sizes n , and that the linear term has a big influence even for larger n .

Theorem 12.1. *For $n \geq 4$, the average number of comparisons in the comparison-optimal dual-pivot quicksort algorithm “Count” when sorting a list of n elements is*

$$\begin{aligned} \mathbb{E}(C_n) &= \frac{9}{5} n H_n - \frac{1}{5} n H_n^{\text{alt}} - \frac{89}{25} n + \frac{67}{40} H_n - \frac{3}{40} H_n^{\text{alt}} - \frac{83}{800} + \frac{(-1)^n}{10} \\ &\quad - \frac{[n \text{ even}]}{320} \left(\frac{1}{n-3} + \frac{3}{n-1} \right) + \frac{[n \text{ odd}]}{320} \left(\frac{3}{n-2} + \frac{1}{n} \right). \end{aligned}$$

From this exact result it is not hard to determine the first few terms of the asymptotic behavior. This is formulated as the following corollary.

Corollary 12.2. *The average number of comparisons in the optimal dual-pivot quicksort algorithm “Count” when sorting a list of n elements is*

$$\mathbb{E}(C_n) = \frac{9}{5}n \log n + An + B \log n + C + \frac{D}{n} + \frac{E}{n^2} + \frac{F[n \text{ even}] + G}{n^3} + O\left(\frac{1}{n^4}\right)$$

with

$$\begin{aligned} A &= \frac{9}{5}\gamma + \frac{1}{5}\log 2 - \frac{89}{25} = -2.3823823670652\dots, & B &= \frac{67}{40} = 1.675, \\ C &= \frac{67}{40}\gamma + \frac{3}{40}\log 2 + \frac{637}{800} = 1.81507227725206\dots, & D &= \frac{11}{16} = 0.6875, \\ E &= -\frac{67}{480} = -0.13958333333333\dots, \\ F &= -\frac{1}{8} = -0.125, & G &= \frac{31}{400} = 0.0775, \end{aligned}$$

asymptotically as n tends to infinity.

Before continuing, let us make a remark on the (non-)influence of the parity of n . It is noteworthy that in Corollary 12.2 no such influence is visible in the first six terms (down to $1/n^2$); only from $1/n^3$ on the parity of n appears. This is somewhat unexpected, since a term $(-1)^n$ appears in Theorem 12.1.

Proof of Theorem 12.1. The partitioning cost and the generating function of strategy “Count” are stated in Proposition 11.1.

We calculate the comparison cost from the partitioning cost by means of Lemma 10.2 and obtain

$$\begin{aligned} C(z) &= -\frac{8\log(1-z)}{5(1-z)^2} + \frac{2\operatorname{artanh}(z)}{5(1-z)^2} - \frac{44}{25(1-z)^2} - \frac{\operatorname{artanh}(z)}{4(1-z)} + \frac{281}{160(1-z)} \\ &\quad + \frac{(1-z)^3}{320}\operatorname{artanh}(z) + \frac{1}{150}z^3 - \frac{27}{1600}z^2 + \frac{17}{1600}z + \frac{3}{800}. \end{aligned}$$

Taking into account that $\operatorname{artanh}(z) = (\log(1+z) - \log(1-z))/2$,

$$\begin{aligned} \sum_{m \geq 1} H_m^{\text{alt}} z^m &= -\frac{\log(1+z)}{(1-z)}, \\ \sum_{m \geq 1} H_m z^m &= -\frac{\log(1-z)}{(1-z)}, \\ \sum_{m \geq 1} m H_m^{\text{alt}} z^m &= z \left(-\frac{\log(1+z)}{(1-z)} \right)' = -\frac{\log(1+z)}{(1-z)^2} + \frac{\log(1+z)}{1-z} + \frac{1}{2(1+z)} - \frac{1}{2(1-z)}, \\ \sum_{m \geq 1} m H_m z^m &= z \left(-\frac{\log(1-z)}{(1-z)} \right)' = -\frac{\log(1-z)}{(1-z)^2} + \frac{1}{(1-z)^2} + \frac{\log(1-z)}{1-z} - \frac{1}{1-z}, \end{aligned}$$

as well as (11.7), we obtain the result. \square

Proof of Corollary 12.2. Insert the expansions of Lemma 8.1 into the explicit representations of Theorem 12.1. \square

Part III. Optimality of the Strategy “Count”

The aim of this section is to show the optimality of the partitioning strategy “Count” in the sense that it minimizes the expected number of key comparisons among all possible partitioning strategies. This is formulated precisely as Theorem 15.2 at the end of this part.

13. INPUT SEQUENCES

Let n be given. For a random permutation (a_1, \dots, a_n) , we use the random variables S , M and L of Section 10 to create a new random variable B whose values are sequences of length $n - 2$ of S letters σ , M letters μ and L letters λ representing the small, medium and large elements in (a_2, \dots, a_{n-1}) , respectively.

We identify B with the random walk $W \in \mathbb{N}_0^3$ starting in the origin where σ , μ and λ correspond to steps e_1 , e_2 and e_3 , respectively. By Lemma 10.1, (S, M, L) and therefore the end point of W are uniformly distributed in Ω' (cf. (10.1)). For each $(s, m, \ell) \in \Omega'$, each such lattice path ending in (s, m, ℓ) is equally likely. Therefore, Lemma 3.1 is applicable.

We will use the notation $|b|$ for the length of the sequence b (above $|b| = n - 2$) and $|b|_\gamma$ for the number of γ 's ($\gamma \in \{\sigma, \mu, \lambda\}$) in the sequence b (above, for example, $|b|_\mu = m$). For $0 \leq t \leq n$, we write $b_{\leq t}$ for the initial segment of length t of b and b_t for the t -th symbol of b .

Lemma 3.1 implies that

$$\mathbb{P}(B_{t+1} = \gamma \mid B_{\leq t} = b_{\leq t}) = \frac{|b_{\leq t}|_\gamma + 1}{t + 3} \quad (13.1)$$

for all sequences b , $0 \leq t < n$ and $\gamma \in \{\sigma, \mu, \lambda\}$.

14. PARTITIONING STRATEGIES

Since we want to compare “Count” with arbitrary partitioning strategies, we must say what such a strategy is in general.

We start with a remark. Partitioning strategies carry out comparisons. As observed in [2], it does not help for saving comparisons to postpone comparing an element with the second pivot if the first comparison was not sufficient to classify the element. Thus we may assume that a strategy always classifies an element completely, either by one or, if necessary, by two comparisons. Thus, strategies we consider classify $n - 2$ elements, one after the other.

A second remark concerns the order in which elements are looked at. While a partitioning strategy can decide in every round which element of the input is looked at next, this decision does not matter for the overall cost (expected number of key comparisons). The reason for this is that the elements not tested up to this point are in random order, so it is irrelevant which of them the strategy chooses for treating next. We use this observation for justifying the assumption that the $n - 2$ elements of a sequence are just read and processed in the order given in the sequence.

Now we can easily describe what a partitioning strategy is. For each sequence of σ 's, μ 's, and λ 's of length smaller than $n - 2$ a strategy specifies whether the next element is to be compared with the smaller pivot p or with the larger pivot q first.

Let $\{\sigma, \mu, \lambda\}^{<n-2}$ denote the set of all sequences over $\{\sigma, \mu, \lambda\}$ of length smaller than $n - 2$.⁵

Definition 14.1. A partitioning strategy for inputs of length n is a function

$$\mathfrak{S}: \{\sigma, \mu, \lambda\}^{<n-2} \rightarrow \{p, q\},$$

where $\mathfrak{S}(\tau) = p$ means that after having seen the initial segment τ of a sequence the next element is compared with p first, similarly for $\mathfrak{S}(\tau) = q$.

Example. Strategy “Count” is given by the function

$$\mathfrak{S}^{\text{ct}}(\tau) = \begin{cases} p & \text{if } |\tau|_{\sigma} \geq |\tau|_{\lambda}, \\ q & \text{otherwise.} \end{cases}$$

Remark 14.2. We know that medium elements, i.e., the μ ’s in these sequences, do not influence the additional comparison count. So one might be tempted to model classification strategies without taking medium elements into account. However, this leads into difficulties, since medium elements encountered underway may influence the decisions made by partitioning procedures (for example, after seeing one σ , a strategy might opt for p , but after seeing one σ and three μ ’s, opt for q). These differences would be hard to take into account if we left out the middle elements from the model.

Note that implicitly the step number is an argument of \mathfrak{S} , so any reaction to the number of steps made so far can also be built into \mathfrak{S} . Nothing would change if it depended on any other parameters or was randomized.

15. OPTIMALITY OF THE STRATEGY “COUNT”

Assume $b = (b_1, \dots, b_{n-2}) \in \{\sigma, \mu, \lambda\}^{n-2}$ is a sequence. Let $b_{\leq t}$ denote the sequence (b_1, \dots, b_t) . We use the terminology *additional cost* of Section 11. A strategy \mathfrak{S} incurs additional cost 1 in step $t + 1$ of the partitioning of b if and only if

- $\mathfrak{S}(b_{\leq t}) = p$ and $b_{t+1} = \lambda$, or
- $\mathfrak{S}(b_{\leq t}) = q$ and $b_{t+1} = \sigma$.

For a strategy \mathfrak{S} and a random sequence B (as in Section 13), let the random variable $A^{\mathfrak{S}}$ be the total number of additional comparisons caused by the sequence B . We are interested in $\mathbb{E}(A^{\mathfrak{S}})$ and show the proposition below, which implies the main result Theorem 15.2.

Proposition 15.1. *Let ct be strategy “Count”, and let \mathfrak{S} be an arbitrary strategy. Then $\mathbb{E}(A^{\text{ct}}) \leq \mathbb{E}(A^{\mathfrak{S}})$.*

Proof. Note that for an arbitrary strategy \mathfrak{S} we have

$$\mathbb{E}(A^{\mathfrak{S}}) = \sum_{\tau \in \{\sigma, \mu, \lambda\}^{<n-2}} \mathbb{P}(\mathfrak{S} \text{ incurs additional cost 1 in step } |\tau| + 1 \mid B_{\leq |\tau|} = \tau) \mathbb{P}(B_{\leq |\tau|} = \tau)$$

⁵Of course, exactly the same information is contained in a classification tree as in [2], if the labels of the elements are ignored. However, the function notation is more convenient here.

by linearity of expectation and the law of total expectation. In view of this formula all we have to do is to show that

$$\begin{aligned} & \mathbb{P}(\text{ct incurs additional cost 1 in step } |\tau| + 1 \mid B_{\leq|\tau|} = \tau) \\ & \leq \mathbb{P}(\mathfrak{S} \text{ incurs additional cost 1 in step } |\tau| + 1 \mid B_{\leq|\tau|} = \tau) \end{aligned} \quad (15.1)$$

for all sequences $\tau \in \{\sigma, \mu, \lambda\}^{<n-2}$.

So assume $1 \leq t < n - 2$ and $\tau \in \{\sigma, \mu, \lambda\}^t$ are given. We work under the assumption that $B_{\leq t} = \tau$.

The first case we consider is that $|\tau|_\sigma \geq |\tau|_\lambda$. Then “Count” chooses to compare with pivot p first, so the probability that it incurs an additional comparison is

$$\frac{|\tau|_\lambda + 1}{t + 3} \quad (15.2)$$

by (13.1). Now consider an arbitrary strategy \mathfrak{S} . The probability that \mathfrak{S} incurs an extra comparison at step $t + 1$ is

$$[\mathfrak{S}(\tau) = p] \cdot \frac{|\tau|_\lambda + 1}{t + 3} + [\mathfrak{S}(\tau) = q] \cdot \frac{|\tau|_\sigma + 1}{t + 3}, \quad (15.3)$$

which is at least as big as (15.2) by the assumption that $|\tau|_\sigma \geq |\tau|_\lambda$. (Note that this argument would apply *mutatis mutandis* if strategy \mathfrak{S} were randomized and chose between p and q by a random experiment.)

The second case $|\tau|_\sigma < |\tau|_\lambda$ is similar. This finishes the proof of Proposition 15.1. \square

Theorem 15.2. *Let $n \geq 2$. The expected number of key comparisons of any partitioning strategy (according to Section 14) when classifying a list of n elements is at least $\mathbb{E}(P_n^{\text{ct}})$ of Proposition 11.1 and this bound is sharp by the partitioning strategy “Count”.*

Reformulated, the partitioning strategy “Count” minimizes the average number of key comparisons.

Proof of Theorem 15.2. Each strategy needs the same number of necessary key comparisons. Proposition 15.1 deals with the additional number of key comparisons. The result follows. \square

REFERENCES

- [1] Jakob Ablinger, *HarmonicSums V1.0*, RISC, 2015, <http://www.risc.jku.at/research/combinatorics/HarmonicSums/>.
- [2] Martin Aumüller and Martin Dietzfelbinger, *Optimal partitioning for dual-pivot quicksort*, ACM Trans. Algorithms **12** (2015), no. 2, Art. 18, 36 pp.
- [3] Philippe Flajolet and Robert Sedgewick, *Analytic combinatorics*, Cambridge University Press, Cambridge, 2009.
- [4] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik, *Concrete mathematics. A foundation for computer science*, second ed., Addison-Wesley, 1994.
- [5] Benjamin Hackl and Daniel Krenn, *Asymptotic expansions in SageMath*, <http://trac.sagemath.org/17601>, 2015, module in SageMath 6.10.beta2.
- [6] P. Hennequin, *Analyse en moyenne d’algorithmes : tri rapide et arbres de recherche*, Ph.D. thesis, École Polytechnique, Palaiseau, 1991.
- [7] Charles A. R. Hoare, *Quicksort*, Comput. J. **5** (1962), no. 1, 10–15.
- [8] Donald E. Knuth, *The art of computer programming. Vol. 3: Sorting and searching*, second ed., Addison-Wesley, Reading, MA, 1998.

- [9] Christian Krattenthaler, *Lattice path enumeration*, Handbook of enumerative combinatorics, Discrete Math. Appl. (Boca Raton), CRC Press, Boca Raton, FL, 2015, pp. 589–678.
- [10] Sri Gopal Mohanty, *Lattice path counting and applications*, Academic Press [Harcourt Brace Jovanovich, Publishers], New York-London-Toronto, Ont., 1979, Probability and Mathematical Statistics.
- [11] The SageMath Developers, *SageMath Mathematics Software (Version 7.4)*, 2016, <http://www.sagemath.org>.
- [12] Carsten Schneider, *Symbolic summation assists combinatorics*, Sémin. Lothar. Combin. **56** (2007), 1–36 (english).
- [13] ———, *EvaluateMultiSums V0.96*, RISC, 2015, Unpublished.
- [14] ———, *Sigma V1.81*, RISC, 2015, <http://www.risc.jku.at/research/combinatorics/Sigma/>.
- [15] Robert Sedgewick, *Quicksort*, Ph.D. thesis, Stanford University, 1975.
- [16] Sebastian Wild, *Java 7's dual pivot quicksort*, Master's thesis, University of Kaiserslautern, 2013, <https://kluedo.ub.uni-kl.de/files/3463/wild-master-thesis.pdf>.
- [17] Sebastian Wild, *Dual-pivot quicksort and beyond: Analysis of multiway partitioning and its practical potential*, Ph.D. thesis, Universität Kaiserslautern, 2016.
- [18] Sebastian Wild, Markus E. Nebel, and Ralph Neininger, *Average case and distributional analysis of dual-pivot quicksort*, ACM Trans. Algorithms **11** (2015), no. 3, Art. 22, 42 pp.
- [19] Vladimir Yaroslavskiy, *Replacement of quicksort in java.util.arrays with new dual-pivot quicksort*, <http://permalink.gmane.org/gmane.comp.java.openjdk.core-libs.devel/2628>, 2009, Archived version of the discussion in the OpenJDK mailing list.

APPENDIX A. PROOF OF LEMMA 10.2

Proof. Multiplying (10.6) by $n(n-1)z^{n-2}$ and summing over all $n \geq 2$ yields

$$\begin{aligned} \sum_{n \geq 2} n(n-1) \mathbb{E}(C_n) z^{n-2} \\ = \sum_{n \geq 2} n(n-1) \mathbb{E}(P_n) z^{n-2} + 6 \sum_{n \geq 1} \sum_{k=0}^{n-1} (n-1-k) z^{n-k-2} \mathbb{E}(C_k) z^k. \end{aligned}$$

Note that the range of the summations has been extended without any consequences because of $\mathbb{E}(C_0) = 0$. We replace $n-1$ by n in the double sum and write it as a product of two generating functions:

$$\begin{aligned} \sum_{n \geq 1} \sum_{k=0}^{n-1} (n-1-k) z^{n-k-2} \mathbb{E}(C_k) z^k &= \sum_{n \geq 0} \sum_{k=0}^n (n-k) z^{n-k-1} \mathbb{E}(C_k) z^k \\ &= \left(\sum_{n \geq 0} n z^{n-1} \right) C(z) = \left(\sum_{n \geq 0} z^n \right)' C(z) = \left(\frac{1}{1-z} \right)' C(z) = \frac{C(z)}{(1-z)^2}. \end{aligned}$$

Thus we obtained

$$C''(z) = P''(z) + \frac{6}{(1-z)^2} C(z)$$

or, equivalently,

$$(1-z)^2 C''(z) - 6C(z) = (1-z)^2 P''(z).$$

Setting $(\theta f)(z) = (1-z)f'(z)$ for a function f , this can be rewritten as

$$((\theta^2 + \theta - 6)C)(z) = (1-z)^2 P''(z).$$

Factoring $\theta^2 + \theta - 6$ as $(\theta-2)(\theta+3)$ and setting $D = (\theta+3)C$, we first have to solve

$$((\theta-2)D)(z) = (1-z)^2 P''(z),$$

i. e.,

$$(1-z)D'(z) - 2D(z) = (1-z)^2 P''(z).$$

Multiplication by $(1-z)$ yields

$$((1-z)^2 D(z))' = (1-z)^3 P''(z).$$

Integration and the fact that $D(0) = C'(0) + 3C(0) = \mathbb{E}(C_1 + 3C_0) = 0$ yields

$$D(z) = \frac{1}{(1-z)^2} \int_0^z (1-s)^3 P''(s) ds.$$

We still have to solve

$$(1-z)C'(z) + 3C(z) = D(z).$$

We multiply by $(1-z)^{-4}$ and obtain

$$((1-z)^{-3}C(z))' = (1-z)^{-4}D(z).$$

As $C(0) = 0$, we obtain

$$C(z) = (1-z)^3 \int_0^z (1-t)^{-4} D(t) dt.$$

□

APPENDIX B. STRATEGY CLAIRVOYANT

Historically, the idea of the strategy “Count” arose from the non-algorithmic strategy “Clairvoyant”, which is described in this section; see also [2]. It will turn out that its cost is only marginally smaller than that of strategy “Count”.

Let s_t and ℓ_t denote the number of elements that have been classified as small and large, respectively, in the first t classification rounds. Set $s_0 = \ell_0 = 0$ and denote the total number of small and large elements by s and ℓ respectively.

Strategy “Clairvoyant”. *When classifying the t -th element, for $1 \leq t \leq n-2$, proceed as follows: If $s - s_{t-1} \geq \ell - \ell_{t-1}$, compare with p first, otherwise compare with q first.*

Note that the strategy “Clairvoyant” cannot be implemented algorithmically, since s and ℓ are not known until the classification is completed.

Instead of up-from-zero situations in the lattice path W_n of Part I (see Corollary 6.2), we have to consider a *down-to-zero situation*. This is a point $(t, 0) \in W_n$ such that $(t-1, 1) \in W_n$. For a randomly (as described in Section 4) chosen path of length n , we have

$$\begin{aligned} \mathbb{E}(\text{number of down-to-zero situations on } W_n) &= \frac{1}{2} (\mathbb{E}(Z_n) - 1) = \frac{1}{2} (H_{n+1}^{\text{odd}} - 1) \\ &= \frac{1}{4} \log n + \frac{\gamma + \log 2 - 2}{4} + \frac{[n \text{ even}] + 1}{4n} - \frac{9[n \text{ even}] + 2}{24n^2} + \frac{[n \text{ even}]}{2n^3} + O\left(\frac{1}{n^4}\right), \end{aligned}$$

asymptotically as n tends to infinity.

Lemma B.1. *Let $n \geq 2$. The expected partitioning cost of strategy “Clairvoyant” is*

$$\begin{aligned} \mathbb{E}(P_n^{\text{cv}}) &= \frac{3}{2}n - \frac{1}{2}H_n^{\text{odd}} - \frac{13}{8} + \frac{3[n \text{ odd}]}{8n} + \frac{[n \text{ even}]}{8(n-1)} \\ &= \frac{3}{2}n - \frac{1}{4} \log n - \frac{1}{4}\gamma - \frac{1}{4} \log 2 - \frac{13}{8} + \frac{1}{8n} + \frac{1}{12n^2} + O\left(\frac{1}{n^3}\right). \end{aligned}$$

The corresponding generating function is

$$\begin{aligned} P^{\text{cv}}(z) = \sum_{n \geq 2} \mathbb{E}(P_n^{\text{cv}}) z^n &= \frac{3}{2(1-z)^2} - \frac{\text{artanh}(z)}{2(1-z)} \\ &\quad - \frac{25z^2}{8(1-z)} + \frac{3+z}{8} \text{artanh}(z) - \frac{3}{2} - \frac{23z}{8}. \end{aligned}$$

Theorem B.2. *For $n \geq 4$, the average number of comparisons in the dual-pivot quicksort algorithm “Clairvoyant” (with oracle) when sorting a list of n elements is*

$$\begin{aligned} \mathbb{E}(C_n^{\text{cv}}) &= \frac{9}{5}nH_n + \frac{1}{5}nH_n^{\text{alt}} - \frac{89}{25}n + \frac{77}{40}H_n + \frac{3}{40}H_n^{\text{alt}} + \frac{67}{800} - \frac{(-1)^n}{10} \\ &\quad + \frac{[n \text{ even}]}{320} \left(\frac{1}{n-3} + \frac{3}{n-1} \right) - \frac{[n \text{ odd}]}{320} \left(\frac{3}{n-2} + \frac{1}{n} \right). \end{aligned}$$

Again, the asymptotic behavior follows from the exact result.

Corollary B.3. *The average number of comparisons in the dual-pivot quicksort algorithm “Clairvoyant” (with oracle) when sorting a list of n elements is*

$$\mathbb{E}(C_n^{\text{cv}}) = \frac{9}{5}n \log n + An + B \log n + C + \frac{D}{n} + \frac{E}{n^2} + \frac{F[n \text{ even}] + G}{n^3} + O\left(\frac{1}{n^4}\right)$$

with

$$\begin{aligned}
A &= \frac{9}{5}\gamma - \frac{1}{5}\log 2 - \frac{89}{25} = -2.6596412392892\dots, & B &= \frac{77}{40} = 1.925, \\
C &= \frac{77}{40}\gamma - \frac{3}{40}\log 2 + \frac{787}{800} = 2.042904116393455\dots, & D &= \frac{13}{16} = 0.8125, \\
E &= -\frac{77}{480} = -0.16041666666666\dots, \\
F &= \frac{1}{8} = 0.125, & G &= -\frac{19}{400} = -0.0475,
\end{aligned}$$

asymptotically as n tends to infinity.

The proof concerning strategy “Clairvoyant” is analogous to the proof of Theorem B.2 and Corollary 12.2. The corresponding generating function is

$$\begin{aligned}
C^{\text{cv}}(z) &= -2\frac{\log(1-z)}{(1-z)^2} - \frac{2\operatorname{artanh}(z)}{5(1-z)^2} - \frac{44}{25(1-z)^2} + \frac{\operatorname{artanh}(z)}{4(1-z)} + \frac{279}{160(1-z)} \\
&\quad - \frac{(1-z)^3}{320}\operatorname{artanh}(z) - \frac{2}{75}z^3 + \frac{123}{1600}z^2 - \frac{113}{1600}z + \frac{13}{800}
\end{aligned}$$

in this case.

APPENDIX C. EMPIRICAL VALIDATION

We implemented strategies “Count” and “Clairvoyant” as dual-pivot quicksort algorithms in a straight-forward way in C++. Pseudocode of the algorithms is presented in Appendix D.

For small input sizes of length $n \in \{2, \dots, 12\}$ we enumerated all permutations of $\{1, \dots, n\}$ and verified that the average number of comparisons (computed over all permutations) obtained from the experimental measurements equals the results from Theorem 12.1 and Theorem B.2, respectively.

For larger inputs, we sorted random permutations of $\{1, \dots, n\}$ for $n = 2^k, k \in \{11, \dots, 28\}$, and counted the comparisons needed to sort the input. For each input size, we sorted 400 different inputs. Figure C.1 shows the measurements we got. From these measurements we conclude that the average comparison counts for sorting over a small number of inputs match the exact average counts from Theorem 12.1 and Theorem B.2, resp., very well.

APPENDIX D. PSEUDOCODE OF DUAL-PIVOT QUICKSORT ALGORITHMS

In this supplementary section, we give the full pseudocode for the strategies “Count” (Algorithm 1) and “Clairvoyant” (Algorithm 2) turned into dual-pivot quicksort algorithms.

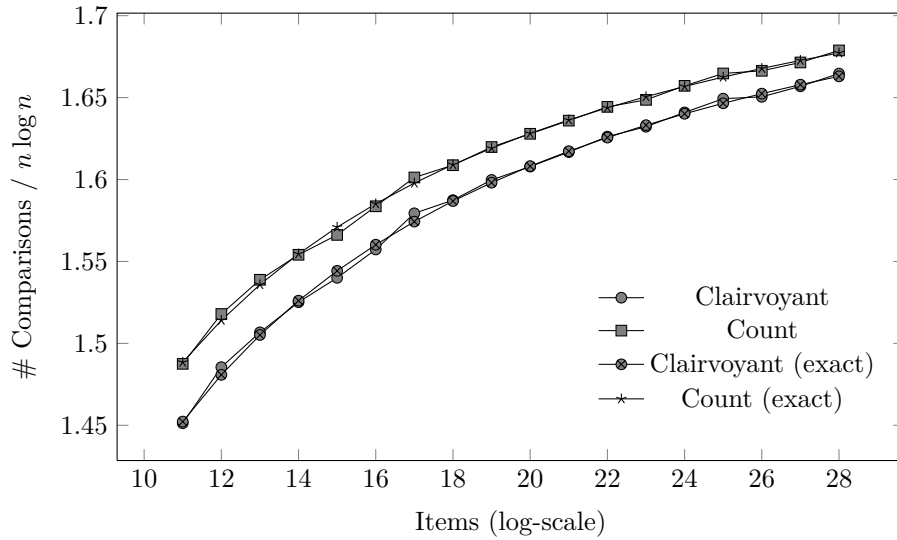


FIGURE C.1. Average comparison count (scaled by $n \log n$) needed to sort a random input of up to 2^{28} integers. We show the measurements we got for the strategies “Count” and “Clairvoyant” together with the exact average comparison counts from Theorem 12.1 and Theorem B.2. All data points from experiments are the average over 400 trials.

MARTIN AUMÜLLER, IT UNIVERSITY OF COPENHAGEN, RUED LANGGAARDS VEJ 7, 2300 COPENHAGEN, DENMARK

E-mail address: `maau@itu.dk`

MARTIN DIETZFELBINGER, FAKULTÄT FÜR INFORMATIK UND AUTOMATISIERUNG, TECHNISCHE UNIVERSITÄT ILMENAU, HELMHOLTZPLATZ 5, 98693 ILMENAU, GERMANY

E-mail address: `martin.dietzfelbinger@tu-ilmenau.de`

CLEMENS HEUBERGER, INSTITUT FÜR MATHEMATIK, ALPEN-ADRIA-UNIVERSITÄT KLAGENFURT, UNIVERSITÄTSSTRASSE 65–67, 9020 KLAGENFURT AM WÖRTHERRSEE, AUSTRIA

E-mail address: `clemens.heuberger@aau.at`

DANIEL KRENN, INSTITUT FÜR MATHEMATIK, ALPEN-ADRIA-UNIVERSITÄT KLAGENFURT, UNIVERSITÄTSSTRASSE 65–67, 9020 KLAGENFURT AM WÖRTHERRSEE, AUSTRIA

E-mail address: `math@danielkrenn.at` or `daniel.krenn@aau.at`

HELMUT PRODINGER, DEPARTMENT OF MATHEMATICAL SCIENCES, STELLENBOSCH UNIVERSITY, 7602 STELLENBOSCH, SOUTH AFRICA

E-mail address: `hproding@sun.ac.za`

Algorithm 1 Dual-Pivot Quicksort Algorithm “Count”**procedure** *Count*(*A*, *left*, *right*)

```

1  if right ≤ left then
2    return
3  if A[right] < A[left] then
4    swap A[left] and A[right]
5  p ← A[left]
6  q ← A[right]
7  i ← left + 1; k ← right − 1; j ← i
8  d ← 0      // d holds the difference of the number of small and large elements.
9  while j ≤ k do
10   if d ≥ 0 then
11     if A[j] < p then
12       swap A[i] and A[j]
13       i ← i + 1; j ← j + 1; d ← d + 1
14     else
15       if A[j] < q then
16         j ← j + 1
17       else
18         swap A[j] and A[k]
19         k ← k − 1; d ← d − 1
20   else
21     if A[k] > q then
22       k ← k − 1; d ← d − 1
23     else
24       if A[k] < p then
25         // Perform a cyclic rotation to the left, i. e.,
26         // tmp ← A[k]; A[k] ← A[j]; A[j] ← A[i]; A[i] ← tmp
27         rotate3(A[k], A[j], A[i])
28         i ← i + 1; d ← d + 1
29       else
30         swap A[j] and A[k]
31       j ← j + 1
32 swap A[left] and A[i − 1]
33 swap A[right] and A[k + 1]
34 Count(A, left, i − 2)
35 Count(A, i, k)
36 Count(A, k + 2, right)

```

Algorithm 2 Dual-Pivot Quicksort Algorithm “Clairvoyant”**procedure** *Clairvoyant*(A , $left$, $right$)

```

1  if  $right \leq left$  then
2    return
3  if  $A[right] < A[left]$  then
4    swap  $A[left]$  and  $A[right]$ 
5   $p \leftarrow A[left]$ 
6   $q \leftarrow A[right]$ 
7   $i \leftarrow left + 1$ ;  $k \leftarrow right - 1$ ;  $j \leftarrow i$ 
8   $d \leftarrow \#(\text{small elements}) - \#(\text{large elements})$            //  $d$  is given by an oracle.
9  while  $j \leq k$  do
10   if  $d \geq 0$  then
11     if  $A[j] < p$  then
12       swap  $A[i]$  and  $A[j]$ 
13        $i \leftarrow i + 1$ ;  $j \leftarrow j + 1$ ;  $d \leftarrow d - 1$ 
14     else
15       if  $A[j] < q$  then
16          $j \leftarrow j + 1$ 
17       else
18         swap  $A[j]$  and  $A[k]$ 
19          $k \leftarrow k - 1$ ;  $d \leftarrow d + 1$ 
20   else
21     if  $A[k] > q$  then
22        $k \leftarrow k - 1$ ;  $d \leftarrow d + 1$ 
23   else
24     if  $A[k] < p$  then
25        $rotate3(A[k], A[j], A[i])$ 
26        $i \leftarrow i + 1$ ;  $d \leftarrow d - 1$ 
27     else
28       swap  $A[j]$  and  $A[k]$ 
29        $j \leftarrow j + 1$ 
30 swap  $A[left]$  and  $A[i - 1]$ 
31 swap  $A[right]$  and  $A[k + 1]$ 
32 Clairvoyant( $A$ ,  $left$ ,  $i - 2$ )
33 Clairvoyant( $A$ ,  $i$ ,  $k$ )
34 Clairvoyant( $A$ ,  $k + 2$ ,  $right$ )

```
